

Technical Report GIT-CC-04-05

Some Layouts Using the Sleepy Stack Approach

Philipp Pfeifferberger, Jun Cheol Park and Vincent J. Mooney III

Center for Research on Embedded Systems and Technology

Georgia Institute of Technology, Atlanta, Georgia, U.S.A.

June 2004

1. Introduction

This technical report elaborates on the methodology and findings presented in “Sleepy Stack Reduction of Leakage Power” by J.C. Park, V. J. Mooney III and P. Pfeifferberger [1]. The scope of this report includes test procedures and data on delay, dynamic and static power for all considered approaches and implementations as well as schematics and layouts for all considered approaches and implementations.

2. Base case

We chose to evaluate the sleepy stack approach on three flavors of gates: an inverter, a full adder and a 4-input multiplexer [1]. These gates were chosen to exemplify a straightforward "memory-like" case (the inverter, which is the basis of SRAM), addition, and a complex gate using NAND, NOR and INV gates (the multiplexer).

The four leakage current reduction approaches considered in this report are compared to a basic CMOS implementation. In all approaches, transistors are placed in two rows, each parallel to continuous Vdd and Gnd contacts. When possible, corresponding Euler paths are chosen from the pull up and pull down networks in the schematic. Using these paths, transistors are placed so that NMOS and PMOS transistors driven by the same input can be connected with a vertical strip of poly and a single contact.

Transistor sizes are specified as a ratio of Width / Length (W/L). In the case of the North Carolina State University (NCSU) [8] design kit targeting the Taiwan Semiconductor Manufacturing Company (TSMC) 0.18 μm process, the smallest possible transistor has a width of 270nm and a length of 180nm, resulting in a ratio of $W/L = 270\text{nm} / 180\text{nm} = 1.5$. This ratio of $W/L=1.5$ signifies the smallest feasible transistor size throughout this report.

Transistors are initially sized so that all circuits have rise and fall times equal to those of an inverter with NMOS $W/L = 1.5$ and PMOS $W/L = 3$. Two of the considered static current reduction approaches explained in Section 3 i.e., the stack [2][3] and sleepy

stack [1] approaches, typically use two transistors each half the width of a particular single transistor in the baseline approach. Since both NOR and INV gates contain transistors with $W/L = 1.5$, the widths of all transistors composing NOR (Appendix D) and INV (Appendix A), as well as NAND (Appendix C) (for uniformity in the multiplexer) are doubled. This doubling is not applied to the Cout' and Sum' circuits of the full adder (Appendix B). The Cout' and Sum' circuits do not contain any minimal width gates when sized to have rise and fall times equal to one inverter and would yield unreasonably large gate sizes if doubled. Schematics for the networks and approaches mentioned in this section can be found in the respective Appendices.

3. Static current reduction approaches

The sleepy stack [1] approach is compared to the base case as well as three established static power reduction techniques: transistor stacking [2][3], Vdd gating via sleep transistors [4] and selective Vdd gating via alternating sleep transistors (the so-called “zigzag” approach) [5]. In order to fairly assess the area needed to implement each approach we chose to always place all transistors in a single line along Vdd and Gnd.

3.1 Stack

The stack approach is implemented by duplicating every transistor in the base case network, with both the original and duplicate bearing half the original transistor width. We chose to always place all transistors in a single row along Vdd and Gnd. Therefore, an increase of the number of transistors and slight decrease in transistor width forces an increase in row length and decrease in row height. E.g., an inverter in the base case (Appendix A.1.b) has a height of 4.7 μm and width of 5.0 μm while an inverter implemented using the stack approach (Appendix A.2.b) has a height of 4.0 μm and width of 6.7 μm .

Creating duplicate transistors in series with the original presents the advantage of maintaining the same layout structure as the base case, with both the original and newly formed transistor gates accessible via the same strip of poly. Appendices A.2.b, B.2.b,

C.2.b and D.2.b illustrate how the base case layout in Appendices A.1.b, B.1.b, C.1.b and D.1.b is largely maintained after applying the stack approach.

3.2 Sleep

For the sleep approach, the transistor sizes of the base case are maintained with added transistors gating the Vdd and Gnd of the circuit. The PMOS sleep transistor between Vdd and the rest of the pull-up network is driven by the Sleep (S) signal, forming a path to Vdd when S is low. The NMOS sleep transistor between Gnd and the rest of the pull-down network is driven by the Sleep' (S') signal, a direct inverse of S. Both of these gating transistors will from hereon out be collectively referred to as “sleep transistors” and will take the width of the largest transistor in their respective base case network.

Vdd and Gnd are disconnected from the circuit when S is high (the circuit is idle), reducing subthreshold leakage but also losing state. Subthreshold leakage can further be reduced by raising the threshold voltage of sleep transistors.

The area penalty incurred with the sleep approach is greater than that of the base case, stack or zigzag approach. In a transistor level layout, additional space is required for sleep transistors as well as S and S' signal lines. Additionally, the gated Vdd/Gnd signal may run between Vdd/Gnd contacts and transistors, further increasing cell height. Layout structure is largely maintained (Appendix B.1.b vs. B.3.b) although the horizontal S and S' lines force use of at least a second metal layer even in simple designs, such as the inverter in Appendix A.3.b.

3.3 Zigzag

The zigzag approach relies on placement of alternating Vdd/Gnd gating transistors in a way that minimizes leakage for a set of most probable input vectors. In order to fairly assess the effectiveness of this approach, the minimal static power dissipated (and associated input vector) is chosen for comparison. In layout, the alternating pull-up / pull-down Vdd/Gnd gating transistors should be placed on abutting ends of adjacent circuits (Appendix A.4.b.). The gating transistors of these abutting circuits allow for a routing scheme similar to that of the sleep approach in Section 3.4. This brings a small but noticeable savings in area over sleep approach.

3.4 Sleepy stack:

The sleepy stack approach combines the stack and sleep approaches by dividing every transistor in the network and placing sleep transistors in parallel with one of the divided transistors [1]. Following the methodology of the sleep approach, sleep transistors are placed on the split transistor closest to the Vdd in the pull up, while the pull down network has sleep transistors placed in parallel with the transistor closest to Gnd. A path from either Gnd or Vdd to output exists in sleep mode, formed by the transistors parallel to the sleep transistors. The sleep transistors reduce resistance when the circuit is switching.

As seen with even simple examples (NAND, Appendix C.1 vs. C.5) the network of the Sleepy stack approach bears little resemblance to that of the base case. In layout, area is increased considerably by the tripling of all transistors. The addition of nodes with an odd number of vertices (by parallel placement of sleep transistors) can lead to elimination of Euler paths and breaks in n and p type regions. As in the stack approach, stacked transistors are accessible by a single contact to a bridging strip of poly (Appendix C.1.b.) Due to placement of the sleep transistor, the stacked transistors can no longer be implemented as one active region with two fingers. As in the sleep approach, sleep signals should be routed horizontally across source/Gnd contacts to conserve space. (Appendix B.5.b.i)

4. Experimental Methodology

Schematics for all models and approaches are created in Cadence Virtuoso Schematic Editor [7] and sized in accordance with the approaches outlined in Section 2. Netlists are extracted from the schematic using Cadence Virtuoso Analog Environment. These netlists are augmented with parameters extracted from the Taiwan Semiconductor Manufacturing Company (TSMC) 0.18 μm process, as well as those of the Berkeley Predictive Technology Model (BPTM) [9] 0.18, 0.13, 0.10 and 0.07 μm processes. The measurements outlined in Sections 4.1, 4.2 and 4.3 were performed using Avant! HSPICE [6]. Unless otherwise specified, input waveforms have a 4 ns period and rise/fall times of 100 ps.

4.1 Delay

Input vectors and input and output triggers are chosen to measure delay across a given circuit's critical path. Fall time is measured as the time between the trigger input edge reaching 50% supply voltage and circuit output edge falling to 50% supply voltage. Likewise, the time between the input edge reaching 50% and circuit output edge rising to 50% supply voltage is recorded as the rise time. This method of measuring delay is carried on throughout all experiments, with only the high/low patterns of the input vectors varying and triggers varying.

4.2 Dynamic power

Dynamic power is measured by asserting clocked semi-random input vectors for a period of 20ns and calculating the average power dissipated during this period.

4.3 Static power

Static power is measured by asserting a set of input vectors as DC sources in HSPICE and measuring the average power dissipated by the circuit during a period of 20 ns.

4.4 Area

Area is measured from full transistor level layouts for the base case as well as stack, sleep, zigzag and sleepy stack approaches. The layouts are created using Cadence Virtuoso layout tool and North Carolina State University's (NCSU's) Cadence design toolkit for 0.18 μm [8]. Layouts are verified with Virtuoso's Digital Rule Checking (DRC) and Layout Versus Schematic (LVS). Because a design kit for sub 0.18 μm sizes is unavailable at time of publication, layout sizes are scaled by ratio of squares with a 10% penalty applied for nonlinear technology components.

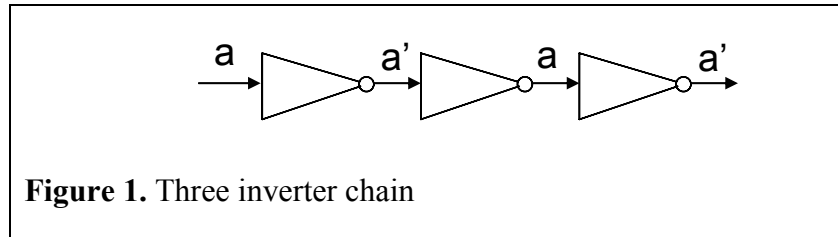
5. Test Circuits

Three test circuits of varying complexity are implemented as described in Sections 2 and 3. A chain of three inverters is chosen as the most basic of logic gates and is indicative of single transistor level behavior and effectiveness. To assess effectiveness using a

complex arrangement of simple logic blocks, a 4-1 MUX is assembled from NAND, NOR and INV gates. Finally, a full adder is chosen as a representation of complex logic, composed of two complex logic gates and two inverters. The effectiveness of the five static power reduction approaches considered in this paper were assessed by applying the experimental methodology of Section 4 to test circuits in Sections 5.1, 5.2 and 5.3.

5.1 Three Inverter chain

Three inverters, equally sized (NMOS $W/L = 3$, PMOS $W/L = 6$ for the base case) are connected in series as shown in Figure 1. Measurements are made across the inverter chain, from the first inverter's input to the last inverter's output.



a. Delay

A square wave is set as input signal for the 3-inverter chain. After four periods, the delay between the input and inverted output is measured.

b. Static power

Static power for the inverter is measured by asserting high and low DC signals and averaging the power dissipated by each input after a period of 20 ns.

c. Dynamic power

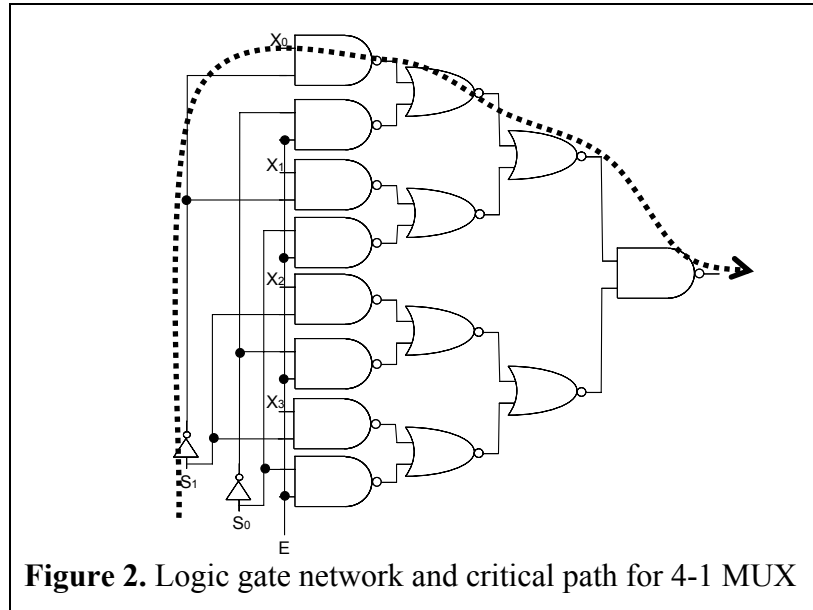
Dynamic power for the inverter is measured by asserting the same square wave used in delay assessment (Section 5.1.a.) to the inverter chain input. Again, the average power dissipated over a period of 20ns is recorded as the Dynamic power of the 3 inverter chain.

d. Area

A full layout for a three inverter chain can be seen in Appendix A.

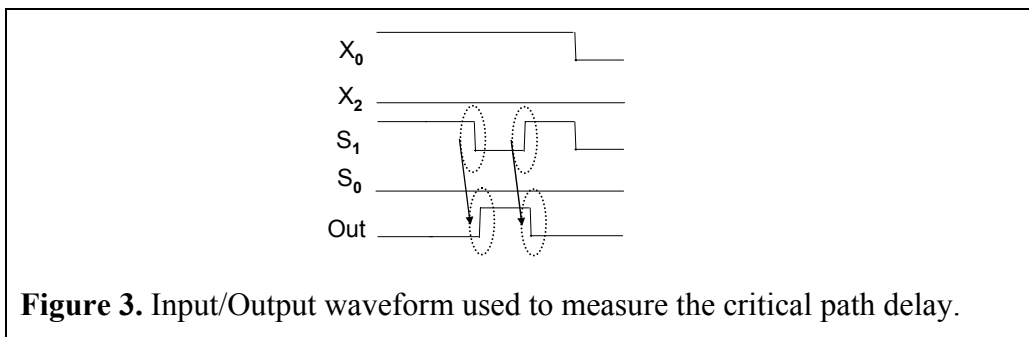
5.2 4-1 Mux

The 4-1 Mux in Figure 2 is implemented using nine 2-input NAND gates, six 2-input NOR gates and two inverters as shown in Figure 2. In the base case, all gates are sized to have rise and fall times equal to an inverter with NMOS $W/L = 3$ and PMOS $W/L = 6$.



a. Delay

The 4-1 Mux delay is measured across the critical path shown in Figure 2, from S_1 (symmetric to S_0) to the output. Before delay is measured across the critical path, the input X_0 is set high and input X_2 is set low. Delay across the critical path is measured by asserting low to S_1 . The output signal is driven high across the path in Figure 2 and the delay is measured as in Figure. 3.



b. Dynamic power

Dynamic power is measured by asserting random values on all inputs for a period of 20ns. The average power during this period is recorded as dynamic power.

c. Static power

From the set of 128 possible inputs, a sample of 8 is chosen and the static power dissipated by the DC signals over a period of 20ns measured. These sample inputs are listed in Table 1.

Table 1. Static power assessment inputs used for 4-1 MUX.

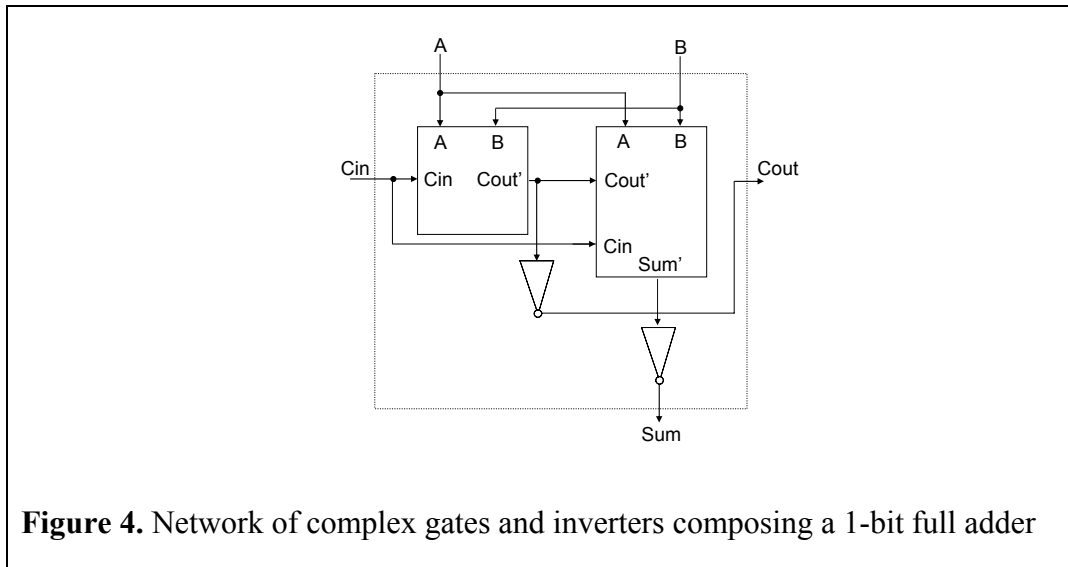
X_0	X_1	X_2	X_3	S_0	S_1	E
0	0	0	0	0	0	0
1	0	0	0	0	0	0
1	0	0	0	0	0	1
1	1	0	0	0	0	1
1	1	0	1	0	1	1
1	1	0	1	1	1	1
1	1	0	1	1	1	0
1	1	1	1	1	1	1

d. Area

The area for a MUX layout is estimated by creating full layouts of components used, i.e. NAND, NOR and INV, and adding the areas of needed components. This sum-of-parts estimation does not take into account extra area needed to wire all components, but the absence of a wiring penalty equally affects all considered approaches. To estimate the area, each component width is multiplied with the height of the tallest component for each approach. For example, if for the stack approach the NOR gate has the largest height, an adjacent inverter would have to use the same source and drain and therefore have an area equal to its base multiplied by the height of the NOR gate.

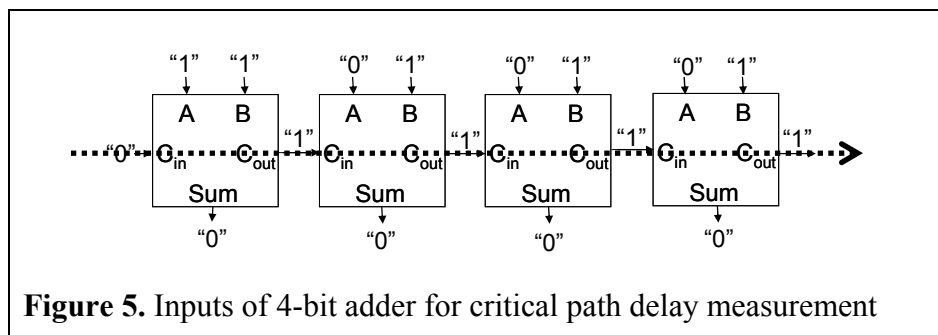
5.3 4-bit Adder

A series of full adders (Figure 4) is created from four logic blocks, one complex logic block that generates inverted Carry out (C_{out}'), one complex logic block that generates an inverted Sum (Sum') and two inverters to create non-inverted signals from the outputs of the two complex blocks. While the inverters are sized to twice the original size (similar to the inverters in Section 5.1), the complex logic blocks are sized to have a rise and fall time equal to an inverter with NMOS $W/L = 1.5$, PMOS $W/L = 3$.



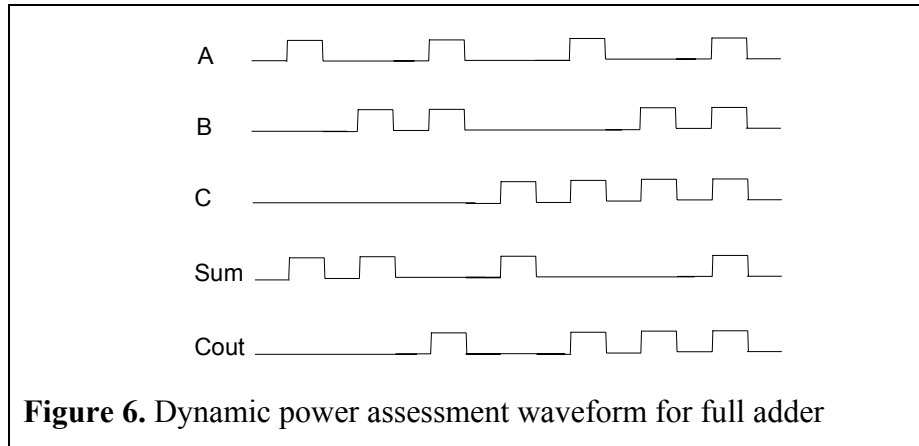
a. Delay

The critical path of the 4-bit adder is formed by the propagation of carry signals. To measure delay, carry propagation is forced across the chain as shown in Figure 5. The delay between adder input signals and the formation of the last C_{out} is taken as the worst case delay.



b. Dynamic power

To assess the dynamic power dissipated by the circuit, a test vector covering every possible input is formed and asserted. A low signal is asserted on all inputs before any high signals to minimize states in which static power is dominant. The resulting waveform (Figure 6) is asserted cyclically for 20ns and the average power dissipated during this period recorded as dynamic power.



c. Static power

All eight possible inputs are in turn imposed as a DC source. The average of the power dissipated for each input after 20ns is recorded as the static power dissipation of the circuit.

d. Area

A full transistor-level layout is created for a 1-bit adder. The area for a four-bit adder is taken as the sum of areas for four 1-bit adders.

5. Conclusions

In terms of area, the sleepy stack approach is better suited for simple logic gates than complex logic gates, as shown in Figure 7. The reason that simple networks are favored by the approach arises out of the sleepy stack structure. The structure of a sleepy stack transistor group consists of the original transistor, stacked duplicate and sleep transistor

connected between the original transistor and stacked duplicate. The center node connecting all three transistors has an odd number of vertices.

Nodes with an odd number of vertices can be included in a Euler path, but only as starting or ending points of the path. Therefore, a Euler path can include at most two sleepy stack transistor groups, forcing separate paths for all other pairs of sleepy stacks. Since continuous active regions depend on Euler paths in the pull-up/pull-down network, the number of separate active regions will be proportional to half the number of sleepy stacked transistors.

The sleepy stack approach could feasibly be implemented in a standard cell library of simple logic blocks. Due to the area penalty, the sleepy stack approach can be better used for applications where static power consumption is critical and cost can be paid in area and delay.

Area for Static Power Reduction Implementations

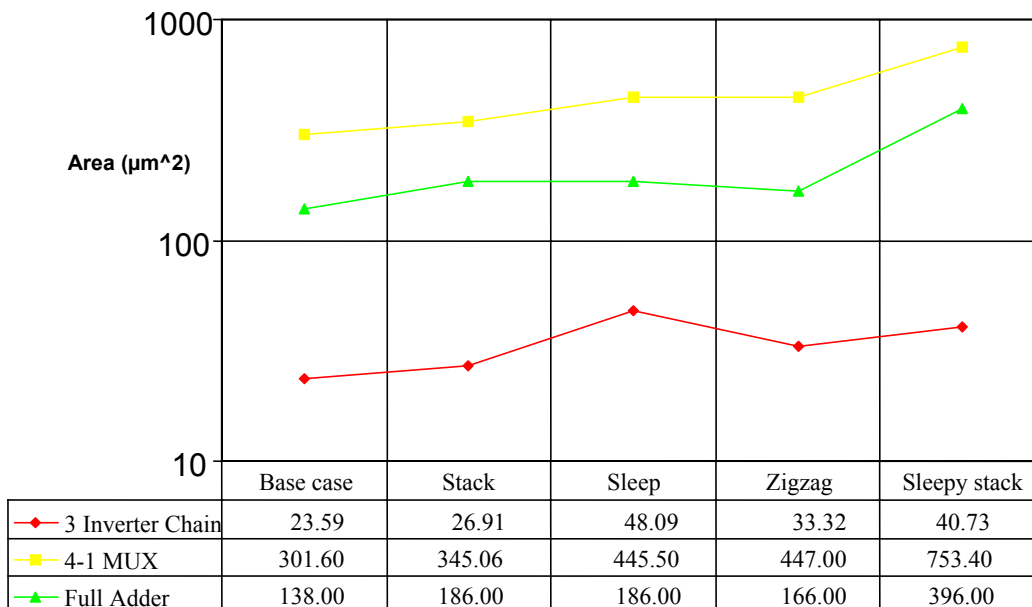


Figure 7. Areas for considered static power reduction implementations and respective test circuits.

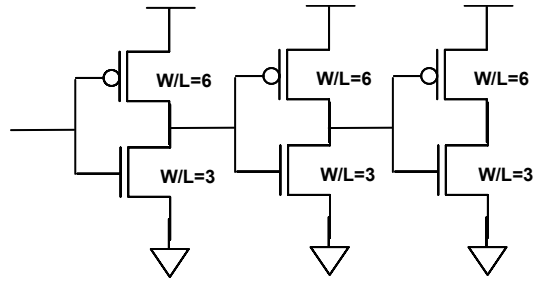
6. References

- [1] J.C. Park, V. J. Mooney III and P. Pfeifferberger, "Sleepy Stack Reduction of Leakage Power," To be published in *PATMOS 2004*
- [2] Z. Chen, M. Johnson, L. Wei and K. Roy, "Estimation of Standby Leakage Power in CMOS Circuits Considering Accurate Modeling of Transistor Stacks," *International Symposium on Low Power Electronics and Design*, pp. 239-244, 1998.
- [3] S. Nadara, S. Borkar, V. De, D. Antoniadis and A. Chandrakasan, "Scaling of Stack Effect and its Application for Leakage Reduction," *International Symposium on Low Power Electronics and Design*, pp. 195-200, August 2001.
- [4] M. Powell, S.-H. Yang, B. Falsafi, K. Roy and T. N. Vijaykumar, "Gated-Vdd: A Circuit Technique to Reduce Leakage in Deep-submicron Cache Memories," *International Symposium on Low Power Electronics and Design*, pp. 90-95, July 2000.
- [5] K.-S. Min, H. Kawaguchi and T. Sakurai, "Zigzag Super Cut-off CMOS (ZSCCMOS) Block Activation with Self-Adaptive Voltage Level Controller: An Alternative to Clock-gating Scheme in Leakage Dominant Era," *IEEE International Solid-State Circuits Conference*, Vol. 1, pp. 400-401, February 2003.
- [6] Avant! Corporation, <http://www.avanticorp.com>.
- [7] Cadence Design Systems, <http://www.cadence.com>.
- [8] NC State University Cadence Tool Information, <http://www.cadence.ncsu.edu>.
- [9] Berkeley Predictive Technology Model (BPTM), <http://www-device.eecs.berkeley.edu/~ptm/>.

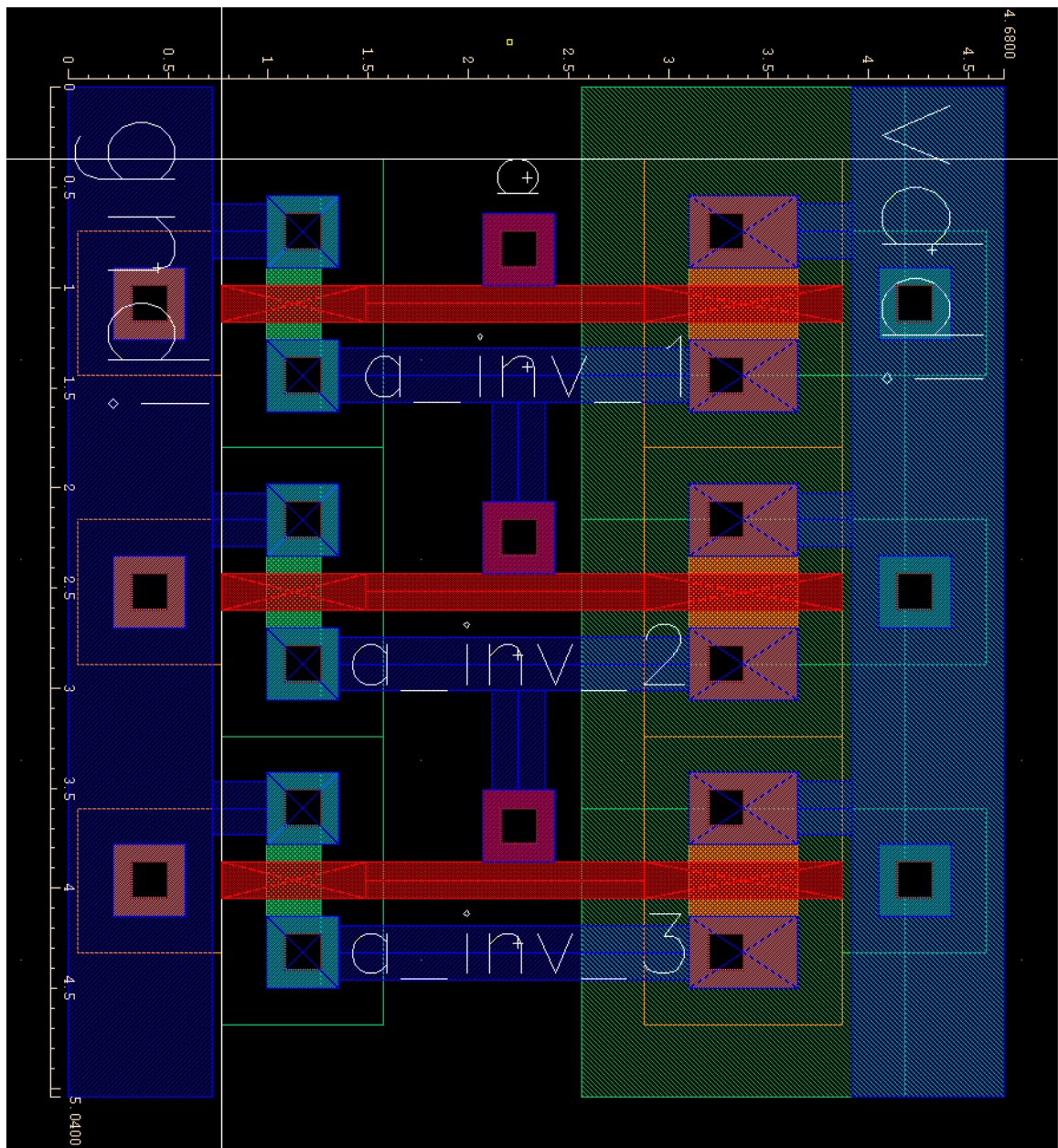
Appendices

- A) 3 Inverter Chain
 - 1) Base approach
 - a) Schematic
 - b) Layout
 - 2) Stack approach
 - a) Schematic
 - b) Layout
 - 3) Sleep approach
 - a) Schematic
 - b) Layout
 - 4) Zigzag approach
 - a) Schematic
 - b) Layout
 - 5) Sleepy stack approach
 - a) Schematic
 - b) Layout
 - 6) 3 inverter chain Data
- B) Full Adder
 - 1) Base approach
 - a) Schematic
 - b) Layout
 - 2) Stack approach
 - a) Schematic
 - (i) Cout'
 - (ii) Sum'
 - b) Layout
 - 3) Sleep approach
 - a) Schematic
 - (i) Cout'
 - (ii) Sum'
 - b) Layout
 - 4) Zigzag approach
 - a) Schematic
 - (i) Cout'
 - (ii) Sum'
 - b) Layout
 - (i) Cout'
 - (ii) Sum'
 - (iii) Full Adder
 - 5) Sleepy stack approach
 - a) Schematic
 - (i) Cout'
 - (ii) Sum'
 - b) Layout

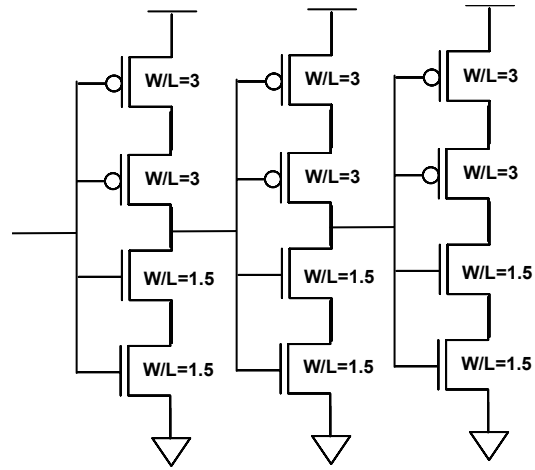
- (i) Cout'
 - (ii) Sum'
 - (iii) Full Adder
- 6) Adder Data
- C) NAND
 - 1) Base case
 - a) Schematic
 - b) Layout
 - 2) Stack approach
 - a) Schematic
 - b) Layout
 - 3) Sleep approach
 - a) Schematic
 - b) Layout
 - 4) Zigzag approach
 - a) Schematic
 - (i) PMOS Sleep
 - (ii) NMOS Sleep
 - b) Layout
 - (i) PMOS Sleep
 - (ii) NMOS Sleep
 - 5) Sleepy stack approach
 - a) Schematic
 - b) Layout
- D) NOR
 - 1) Base case
 - a) Schematic
 - b) Layout
 - 2) Stack approach
 - a) Schematic
 - b) Layout
 - 3) Sleep approach
 - a) Schematic
 - b) Layout
 - 4) Zigzag approach
 - a) Schematic
 - (i) PMOS Sleep
 - (ii) NMOS Sleep
 - b) Layout
 - (i) PMOS Sleep
 - (ii) NMOS Sleep
 - 5) Sleepy stack approach
 - a) Schematic
 - b) Layout
- E) MUX Data



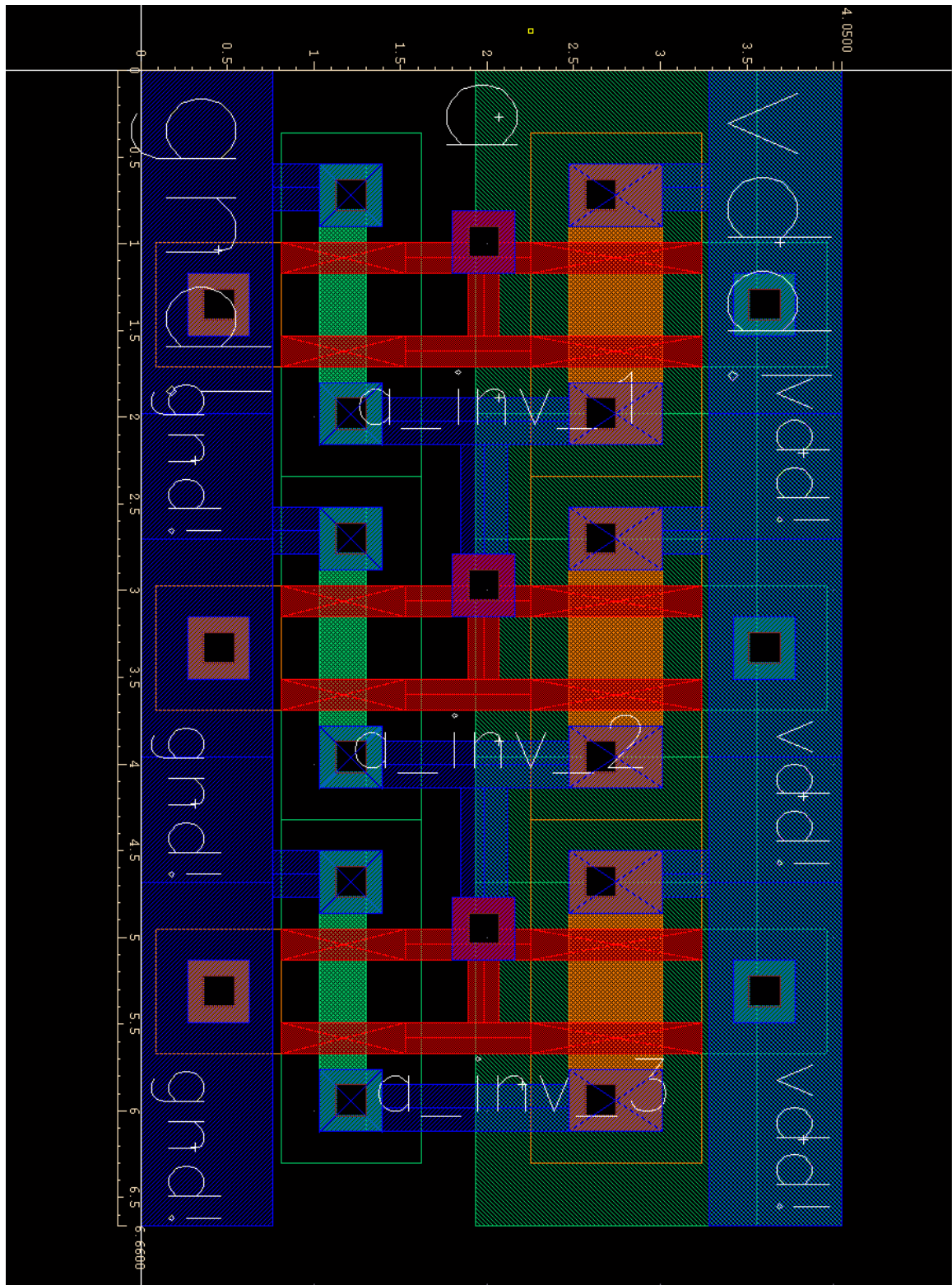
A.1.a. Base approach 3 inverter chain schematic



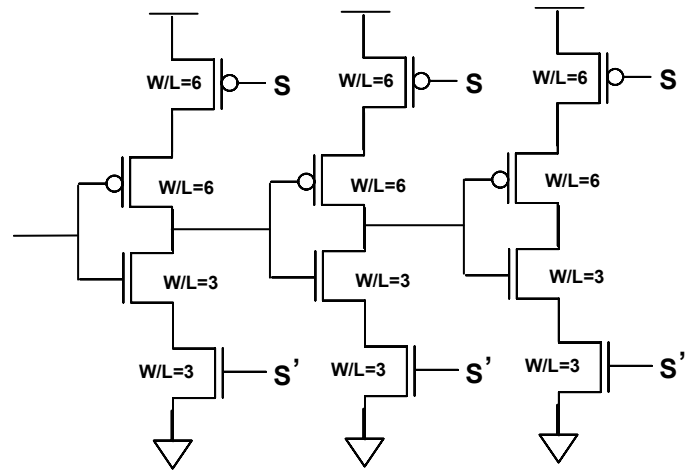
A.1.b. Base approach 3 inverter chain layout



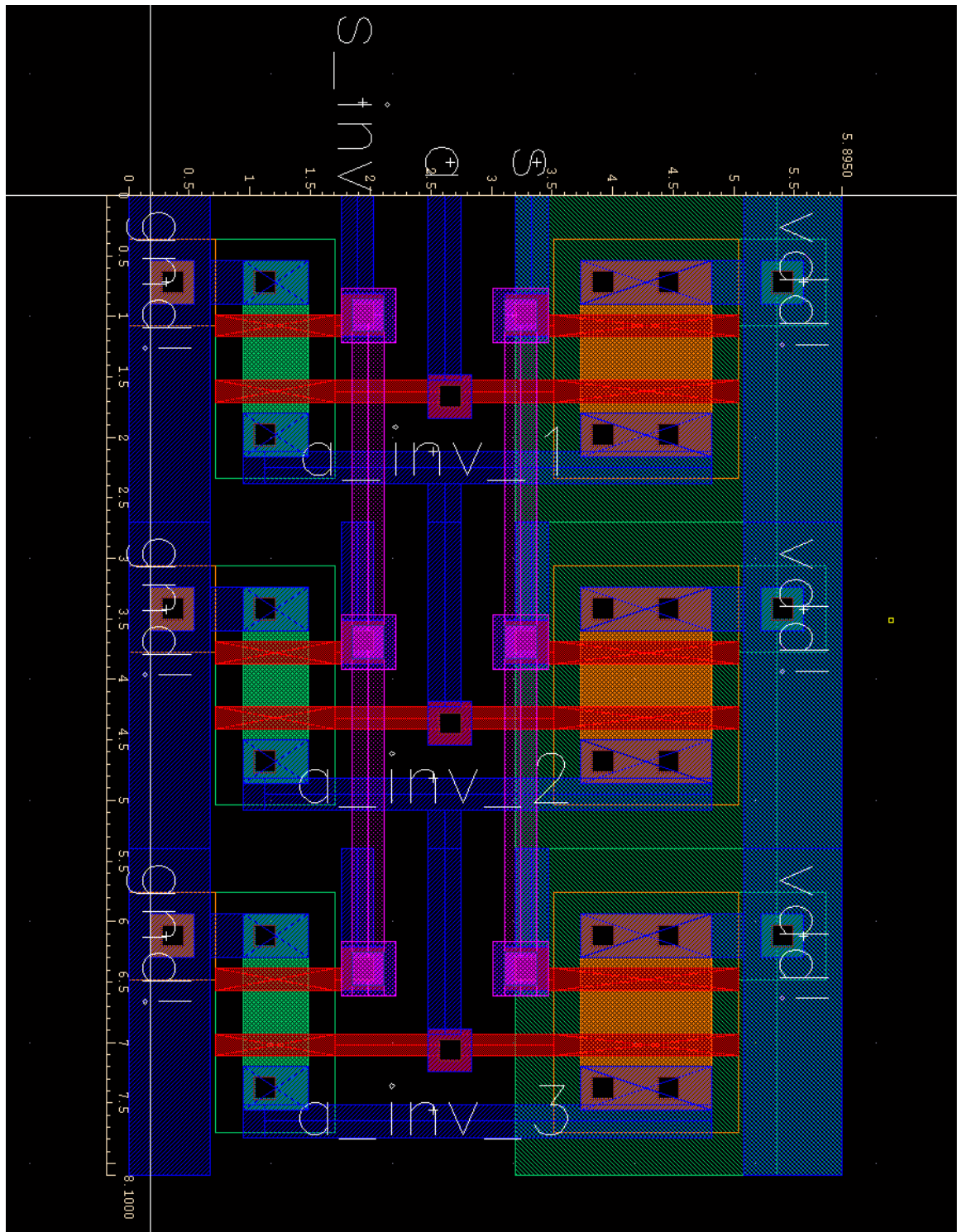
A.2.a. Stack approach 3 inverter chain schematic



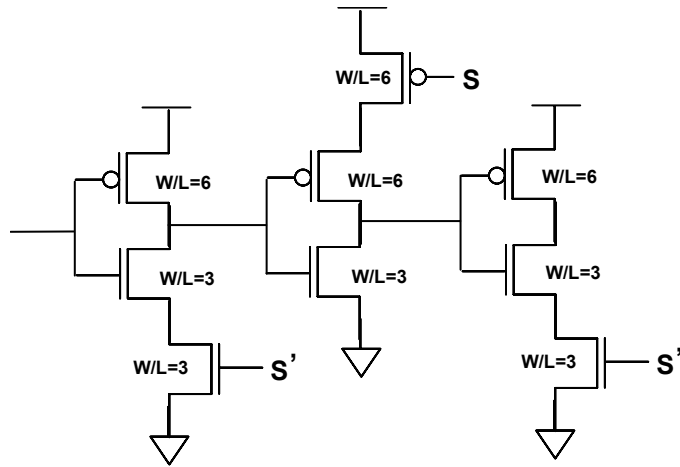
A.2.b. Stack approach 3 inverter chain layout



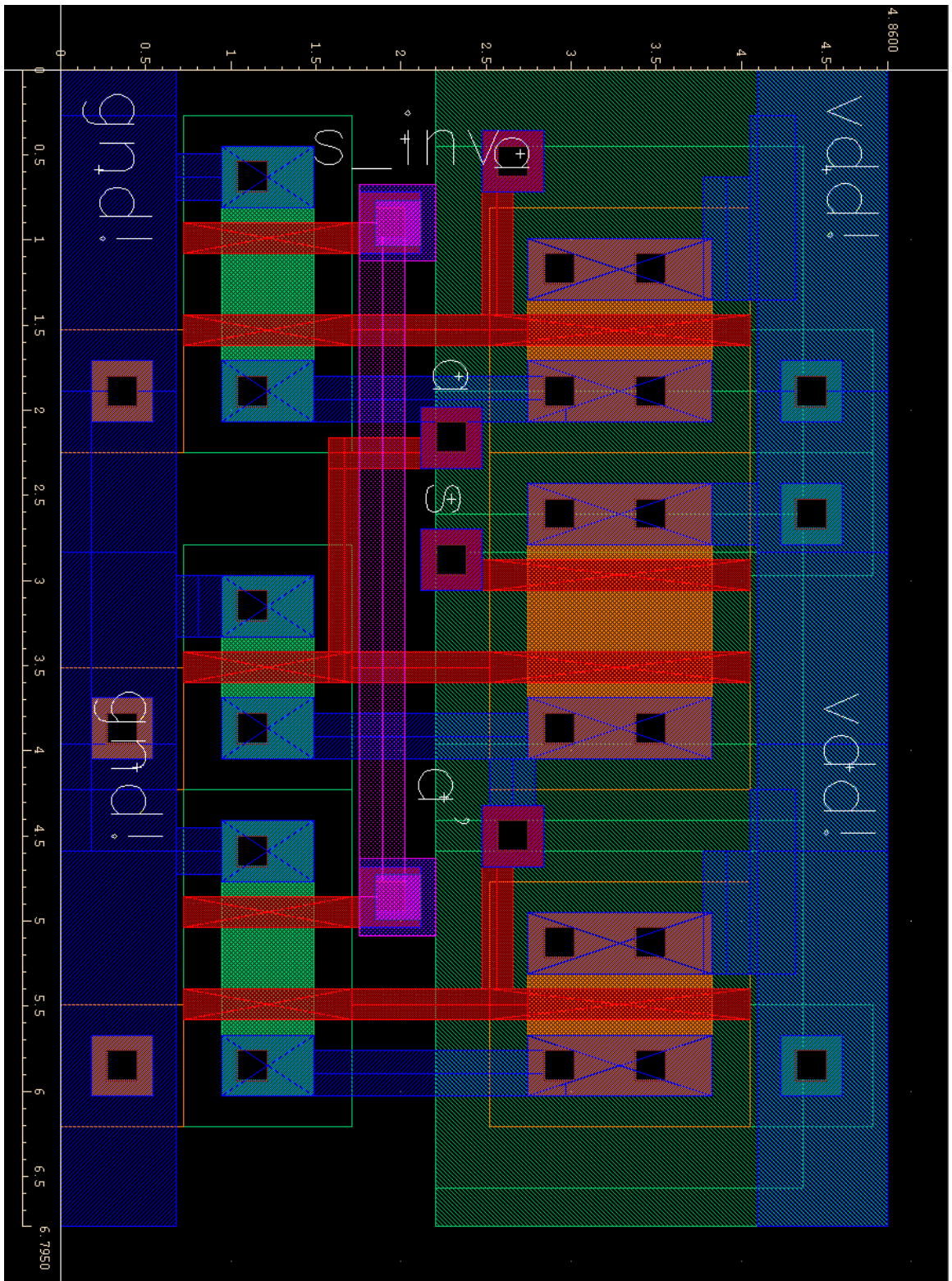
A.3.a. Sleep approach 3 inverter chain schematic



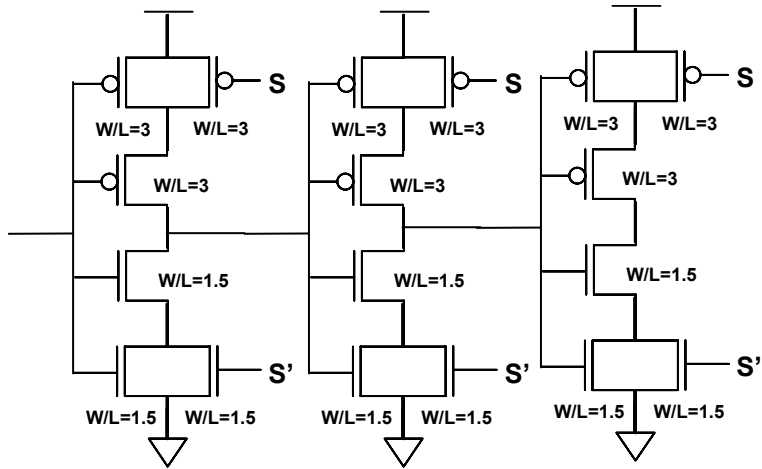
A.3.b. Sleep approach 3 inverter chain layout



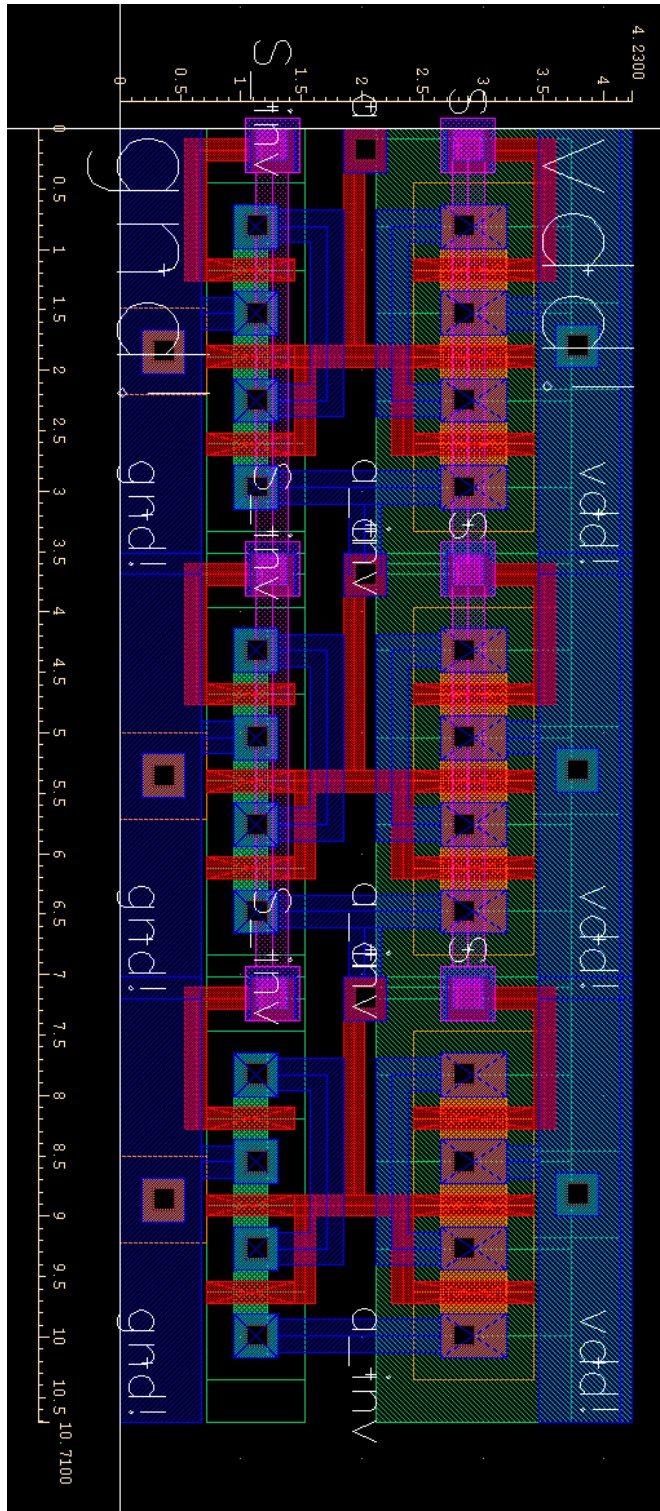
A.4.a. Zigzag approach 3 inverter chain schematic



A.4.b. Zigzag approach 3 inverter chain layout



A.5.a. Sleepy stack approach 3 inverter chain schematic



A.5.b. Sleepy stack approach 3 inverter chain layout

TSMC 0.18 μ	Propagation delay (s)	Static Power (W)	Dynamic Power (W)	Area (μ^2)
Base case	9.56E-11	4.50E-11	3.16E-06	23.59
Stack	2.46E-10	8.99E-12	3.20E-06	26.91
Sleep	1.56E-10	1.44E-11	4.79E-06	48.09
ZigZag	1.34E-10	5.63E-12	5.43E-06	33.32
Sleepy Stack	1.78E-10	1.64E-11	3.46E-06	40.73
Sleep (dual Vth)	2.22E-10	1.09E-12	4.56E-06	48.09
ZigZag (dual Vth)	1.76E-10	1.06E-17	5.21E-06	33.32
Sleepy Stack (dual Vth)	2.19E-10	5.96E-16	3.18E-06	40.73

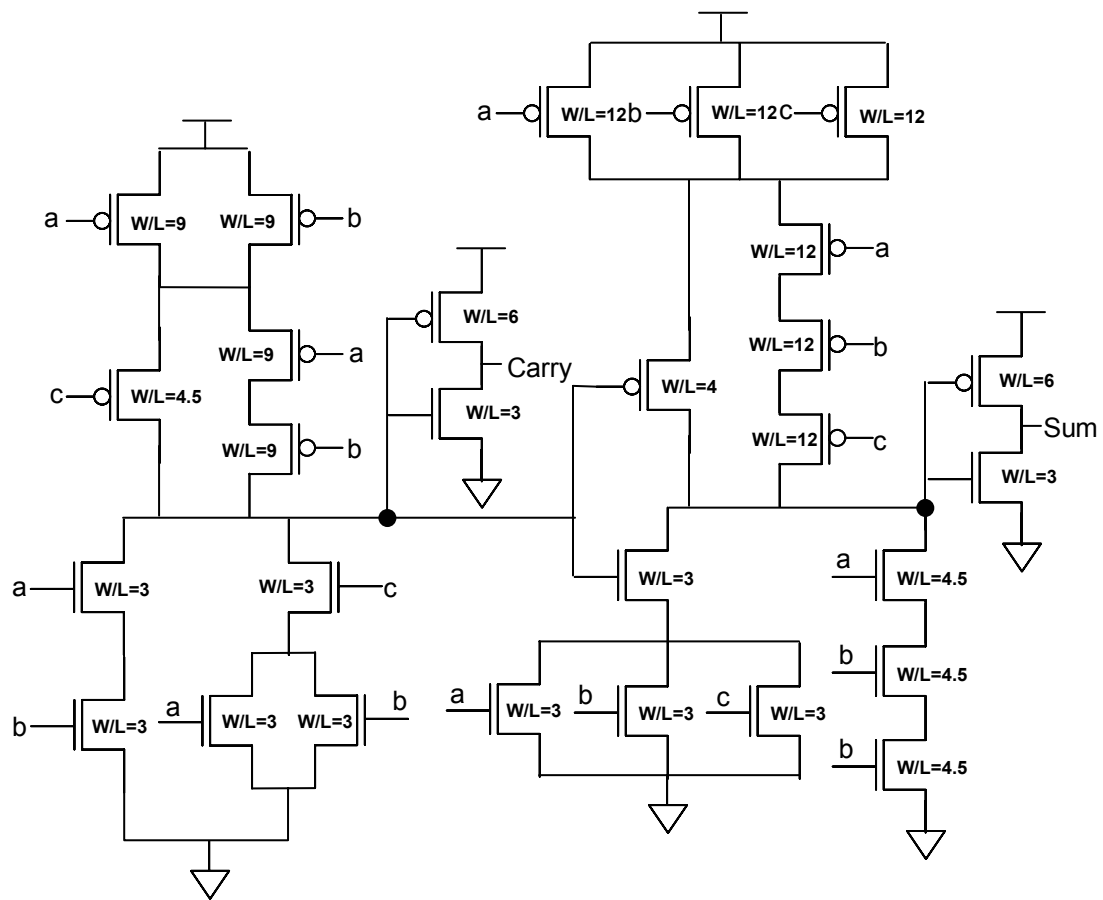
Berkeley 0.18 μ	Propagation delay (s)	Static Power (W)	Dynamic Power (W)	Area (μ^2)
Base case	7.73E-11	1.70E-09	4.94E-06	23.59
Stack	1.95E-10	2.31E-10	3.63E-06	26.91
Sleep	1.06E-10	5.48E-10	7.79E-06	48.09
ZigZag	1.01E-10	3.31E-10	8.69E-06	33.32
Sleepy Stack	1.38E-10	4.05E-10	4.85E-06	40.73
Sleep (dual Vth)	1.55E-10	1.11E-12	6.83E-06	48.09
ZigZag (dual Vth)	1.47E-10	4.14E-16	8.04E-06	33.32
Sleepy Stack (dual Vth)	1.87E-10	4.99E-14	3.99E-06	40.73

Berkeley 0.13 μ	Propagation delay (s)	Static Power (W)	Dynamic Power (W)	Area (μ^2)
Base case	7.00E-11	1.48E-09	2.15E-06	13.54
Stack	1.70E-10	1.00E-10	1.56E-06	15.44
Sleep	9.34E-11	2.64E-10	3.21E-06	27.59
ZigZag	8.14E-11	2.32E-10	4.03E-06	19.12
Sleepy Stack	1.20E-10	1.82E-10	2.03E-06	23.37
Sleep (dual Vth)	1.41E-10	6.73E-13	2.62E-06	27.59
ZigZag (dual Vth)	1.07E-10	8.92E-15	3.50E-06	19.12
Sleepy Stack (dual Vth)	1.64E-10	1.75E-13	1.77E-06	23.37

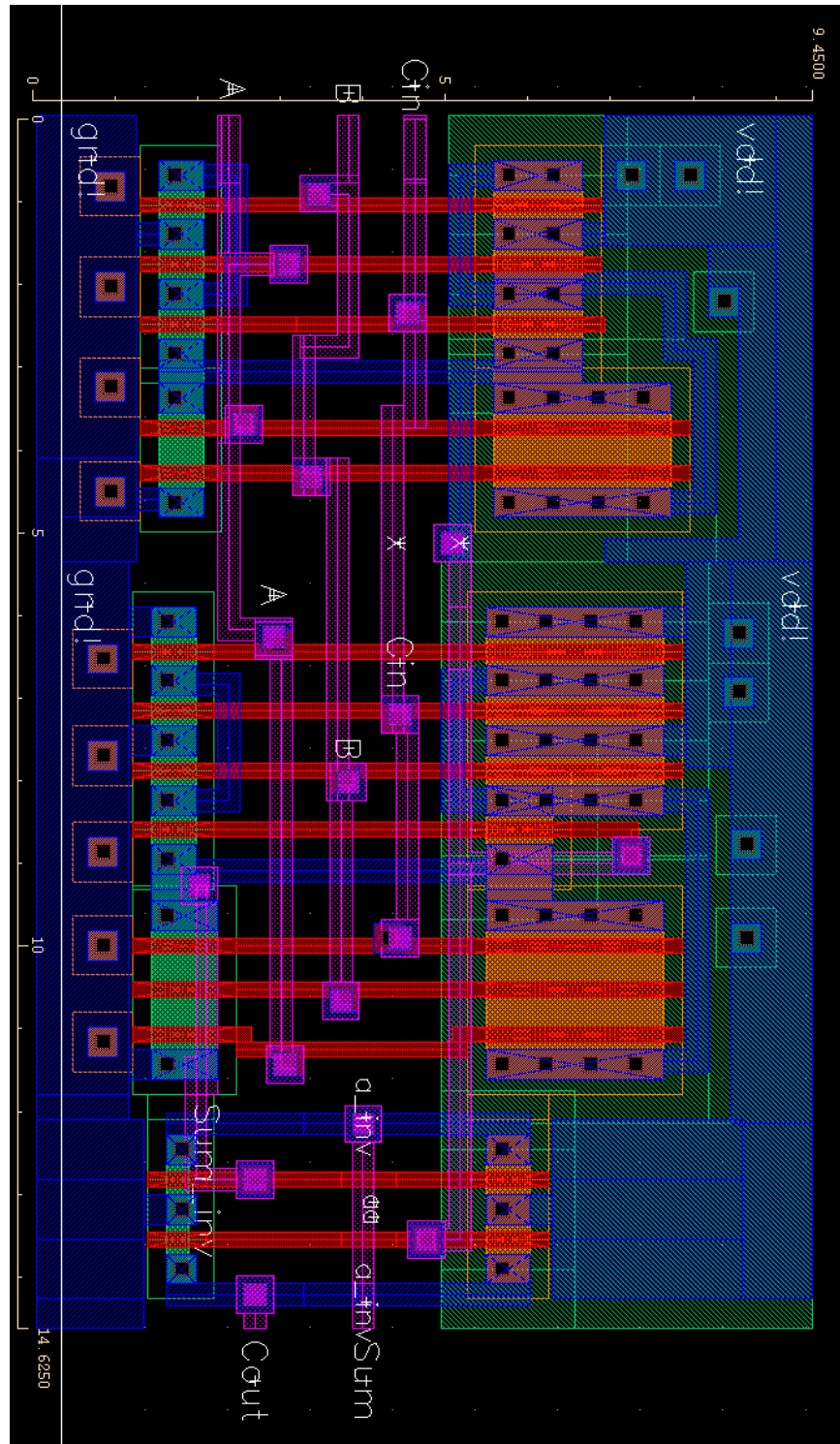
Berkeley 0.10 μ	Propagation delay (s)	Static Power (W)	Dynamic Power (W)	Area (μ^2)
Base case	5.36E-11	6.74E-09	1.67E-06	8.01
Stack	1.30E-10	2.87E-10	1.05E-06	9.14
Sleep	7.05E-11	6.77E-10	2.66E-06	16.33
ZigZag	6.21E-11	5.40E-10	2.80E-06	11.31
Sleepy Stack	9.28E-11	5.39E-10	1.60E-06	13.83
Sleep (dual Vth)	1.02E-10	5.39E-13	2.15E-06	16.33
ZigZag (dual Vth)	8.28E-11	3.44E-14	2.68E-06	11.31
Sleepy Stack (dual Vth)	1.22E-10	5.18E-13	1.17E-06	13.83

Berkeley 0.07 μ	Propagation delay (s)	Static Power (W)	Dynamic Power (W)	Area (μ^2)
Base case	4.61E-11	1.24E-08	6.56E-07	3.92
Stack	1.28E-10	9.89E-10	4.08E-07	4.48
Sleep	6.98E-11	2.40E-09	9.49E-07	8.00
ZigZag	5.99E-11	2.27E-09	1.05E-06	5.54
Sleepy Stack	8.75E-11	1.77E-09	6.35E-07	6.78
Sleep (dual Vth)	1.14E-10	4.32E-13	8.58E-07	8.00
ZigZag (dual Vth)	9.03E-11	3.84E-13	9.87E-07	5.54
Sleepy Stack (dual Vth)	1.38E-10	9.88E-13	4.88E-07	6.78

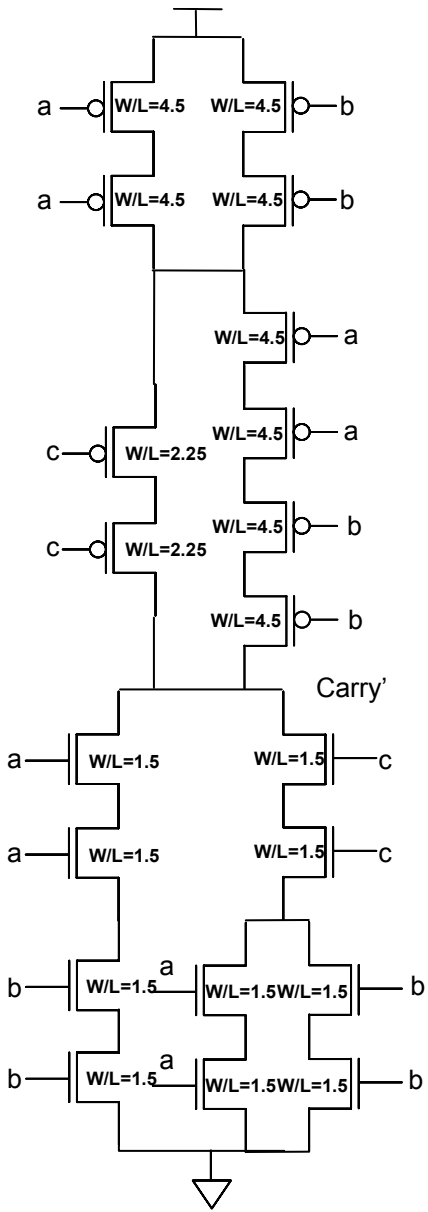
A.6. 3 inverter chain data



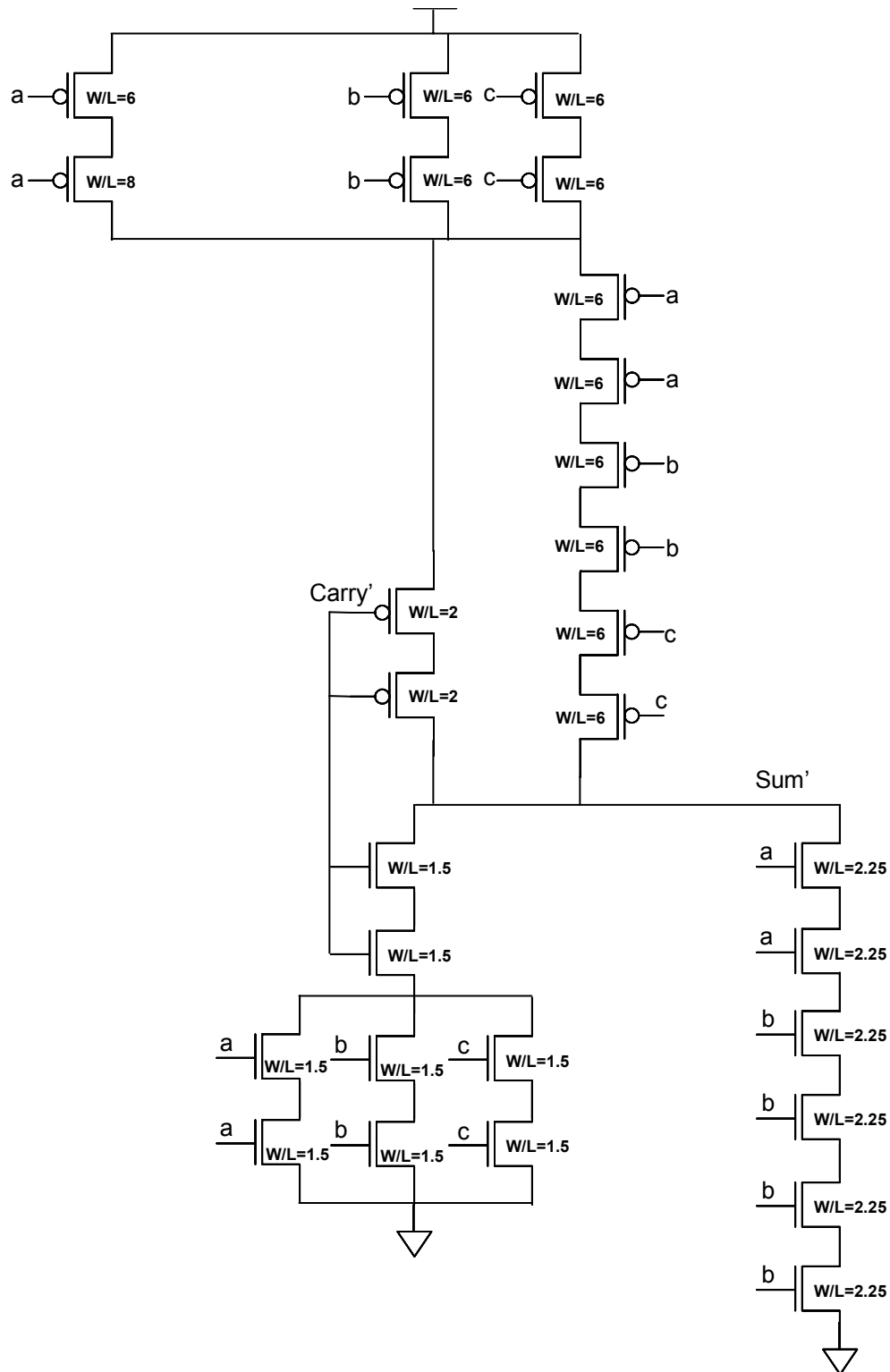
B.1.a. Base case full adder schematic



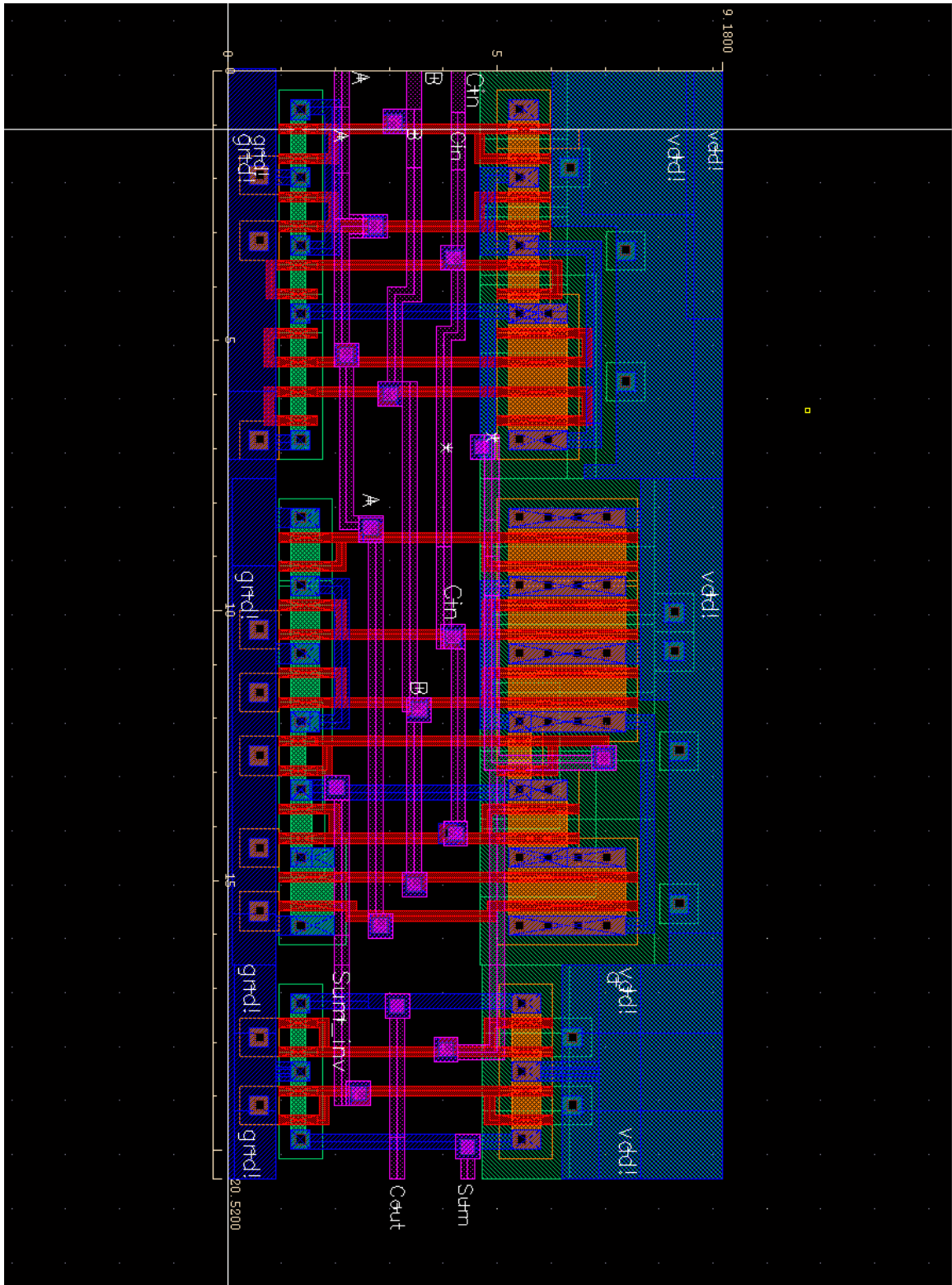
B.1.b. Base case full adder layout



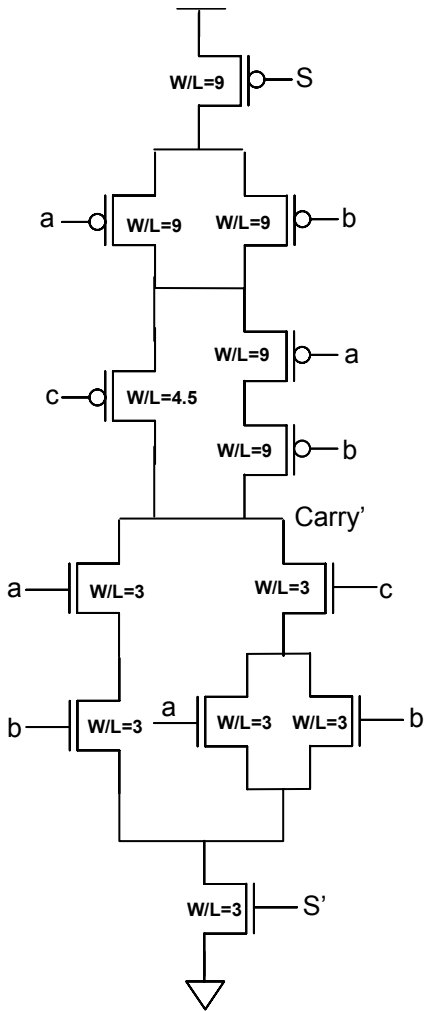
B.2.a.i. Stack approach Full Adder Cout' schematic



B.2.a.ii. Stack approach Full Adder Sum' schematic

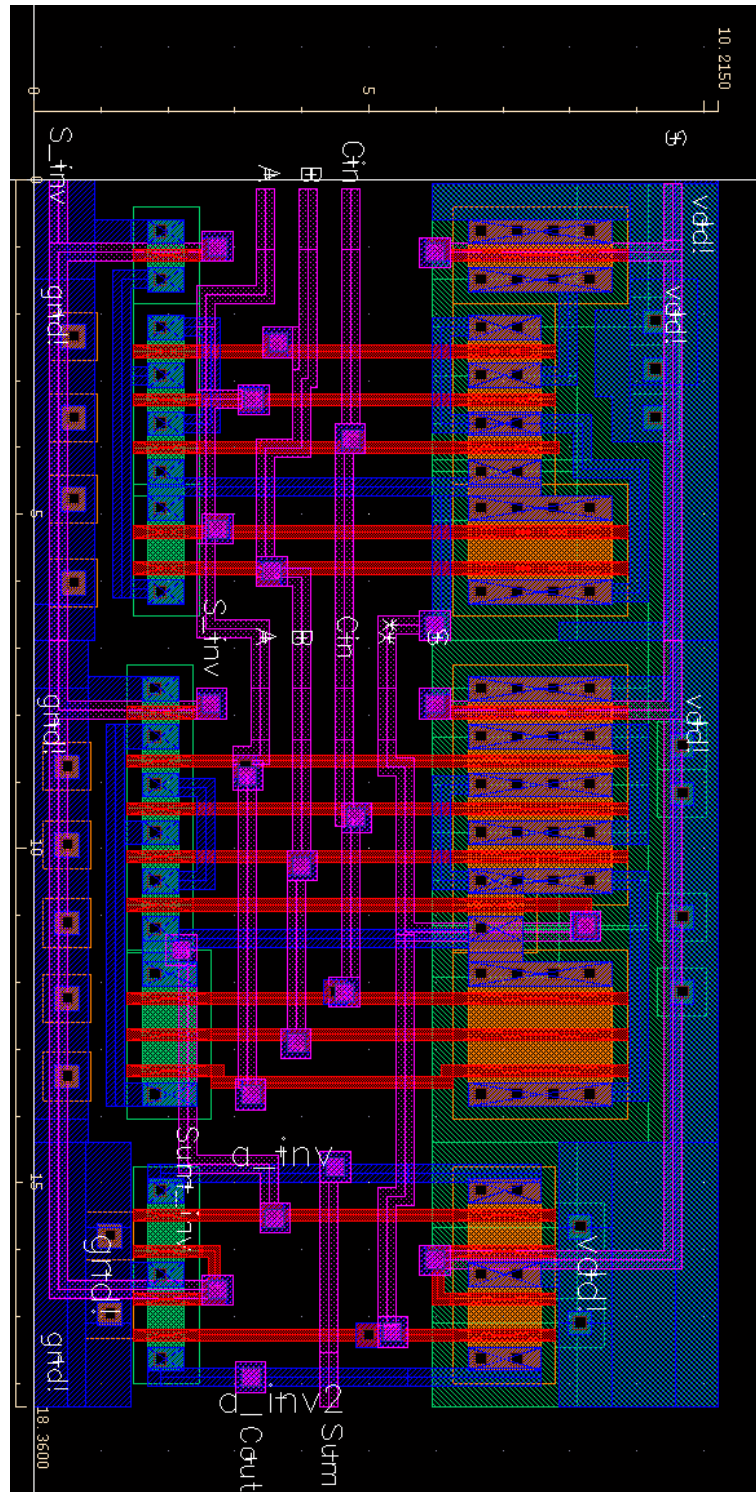


B.2.b. Stack approach Full Adder layout

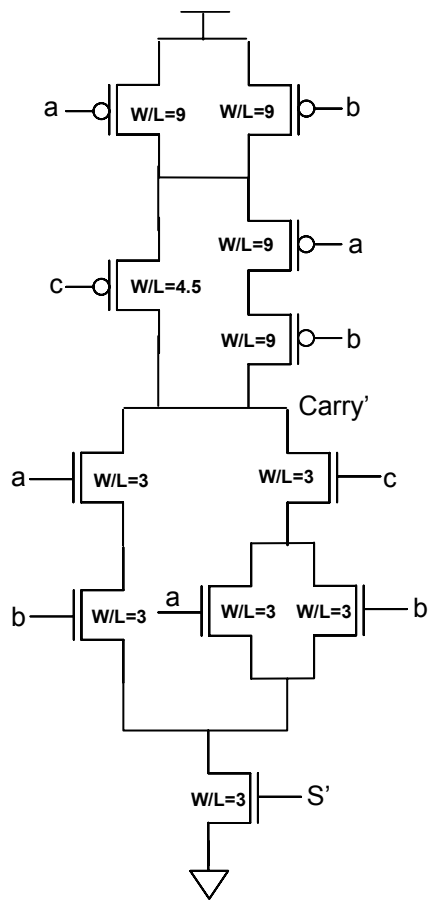


B.3.a.i. Sleep approach Full Adder Cout' schematic

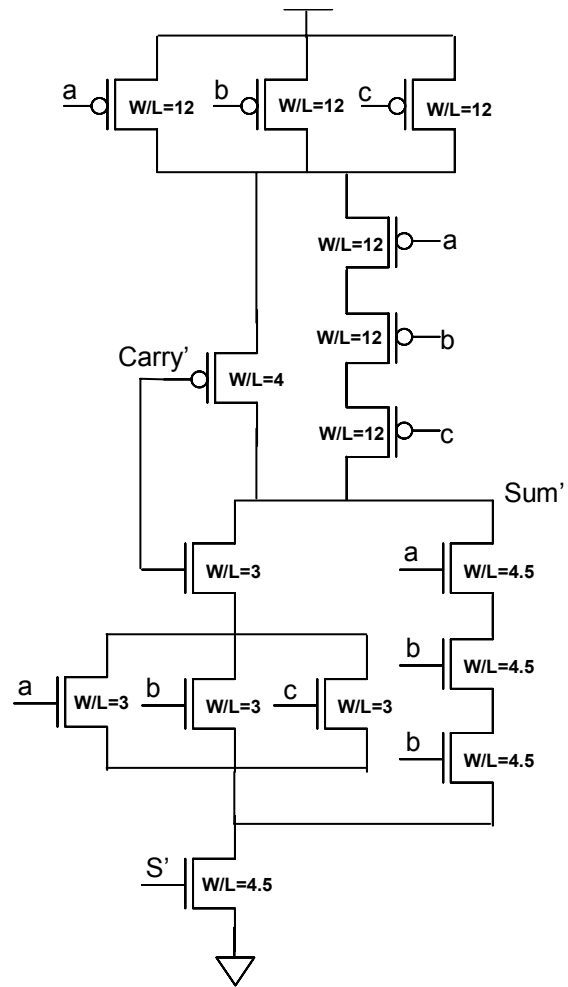




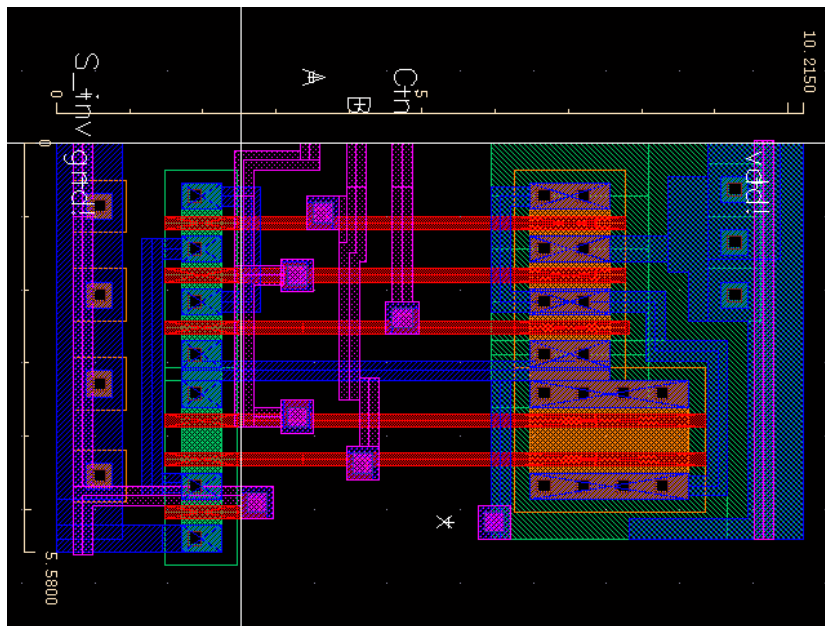
B.3.b. Sleep approach Full Adder layout



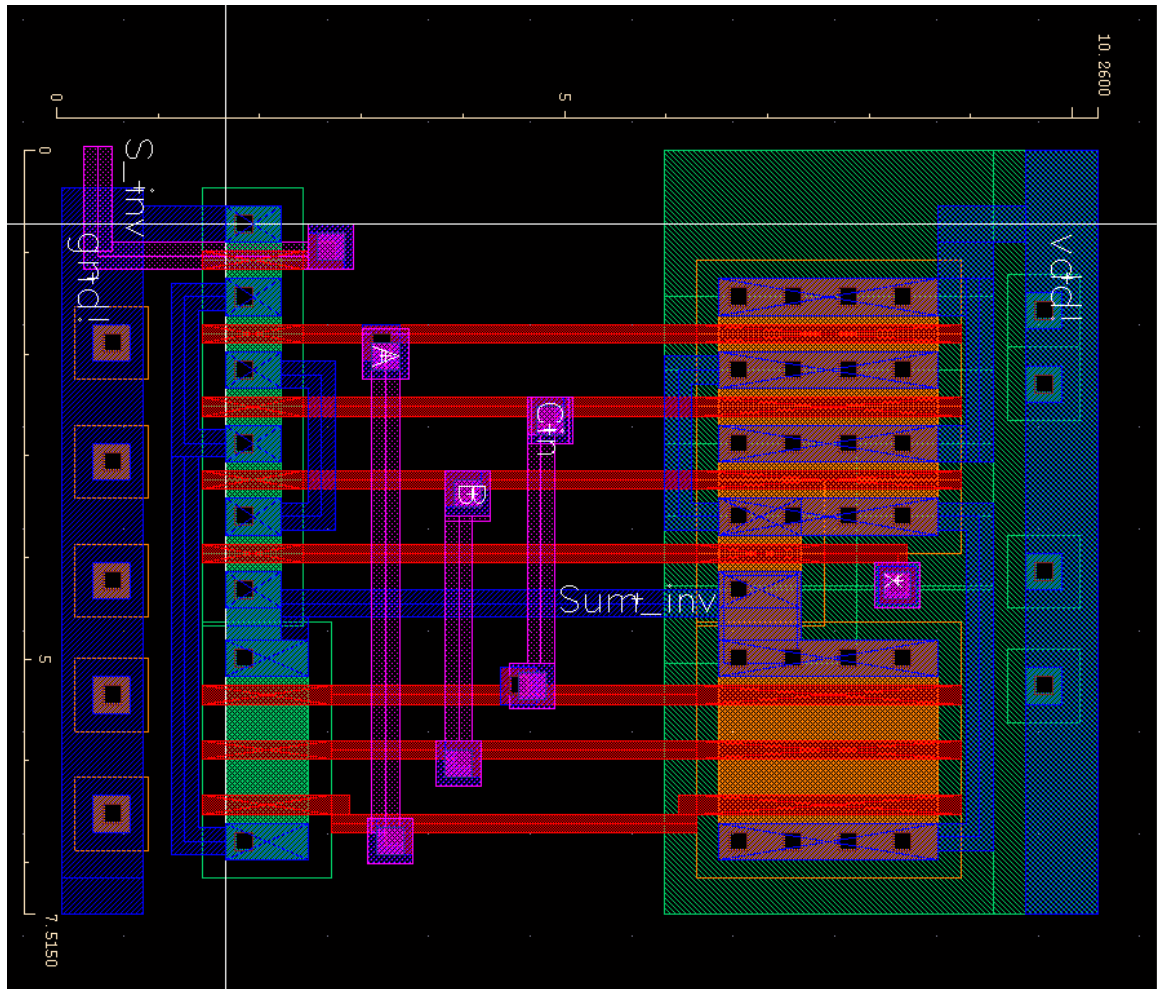
B.4.a.i. Zigzag approach Full Adder Cout' schematic.



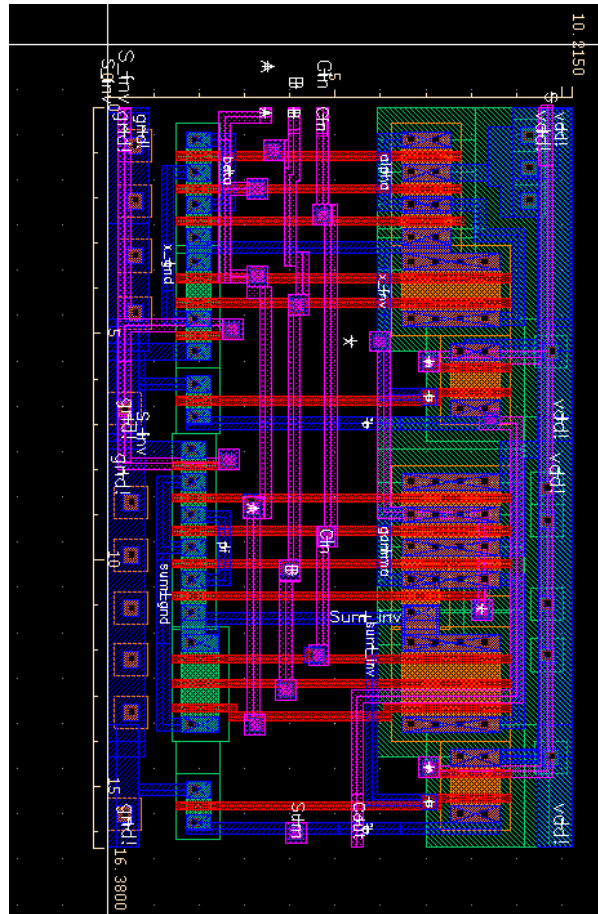
B.4.a.ii. Zigzag approach Full Adder Sum' schematic



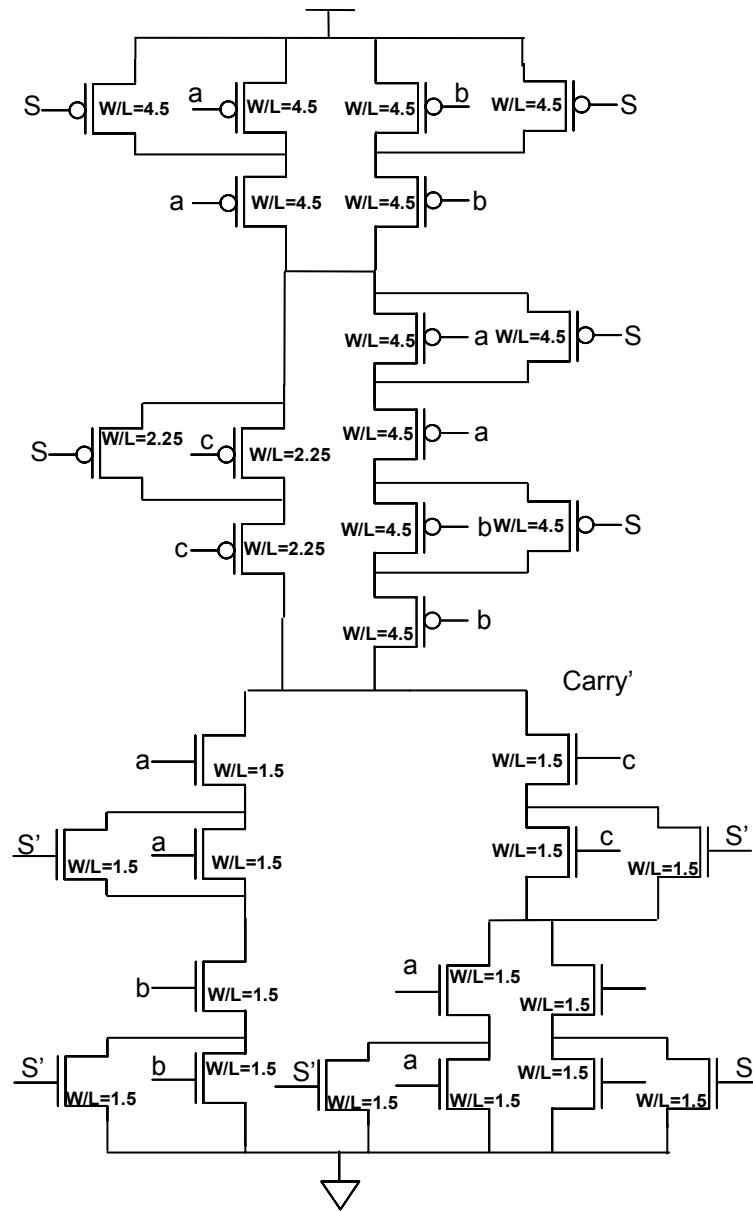
B.4.b.i. Zigzag approach Full Adder Cout' layout



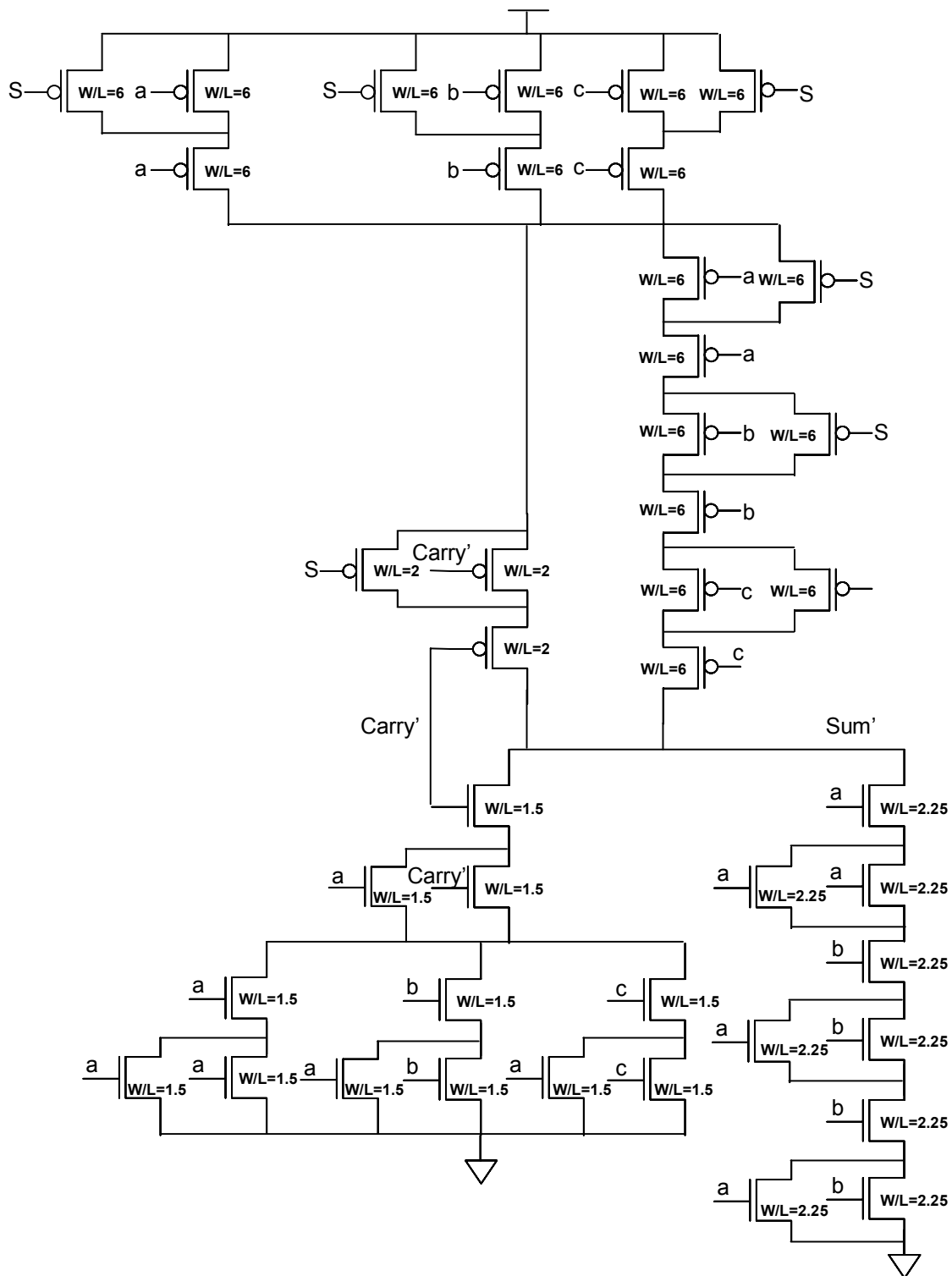
B.4.b.ii. Zigzag approach Full Adder Cout' layout



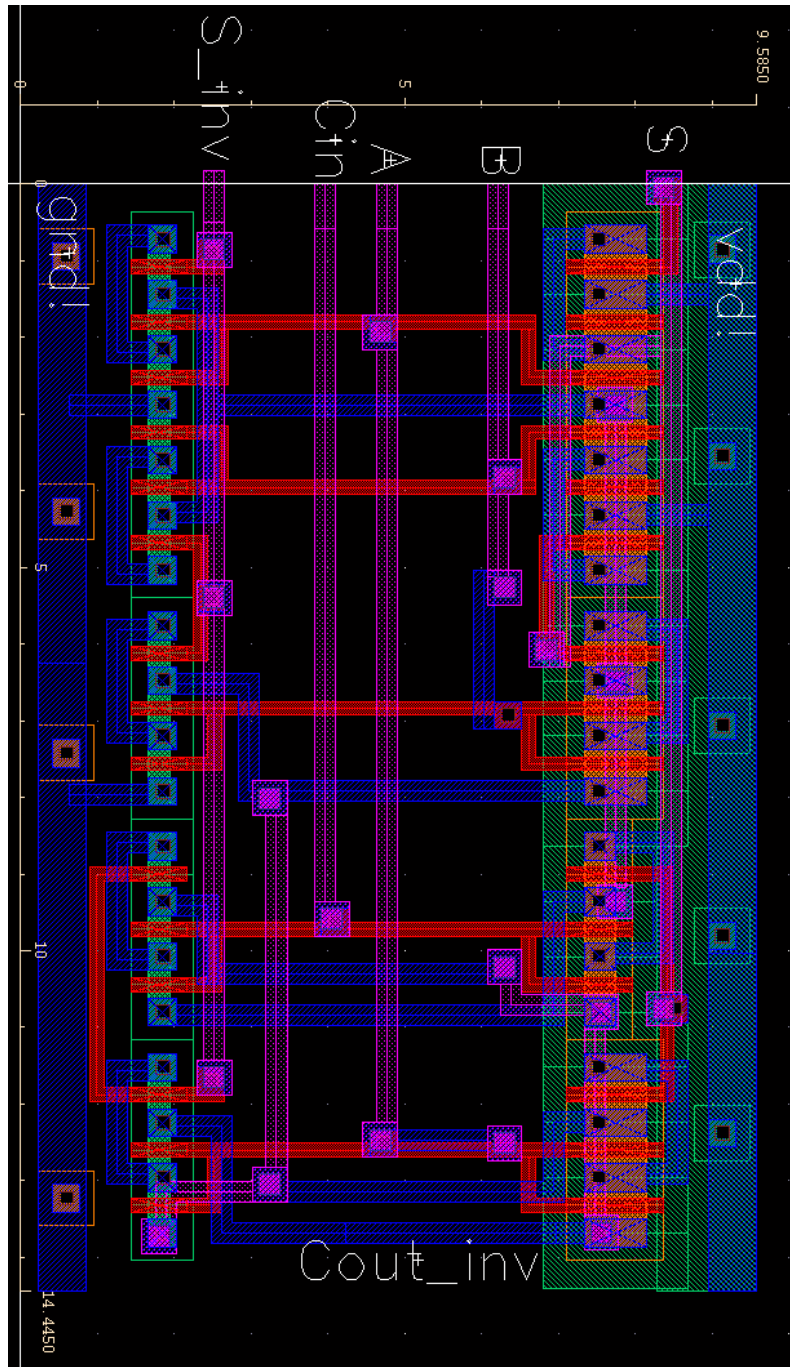
B.4.b.iii. Zigzag approach Full Adder layout



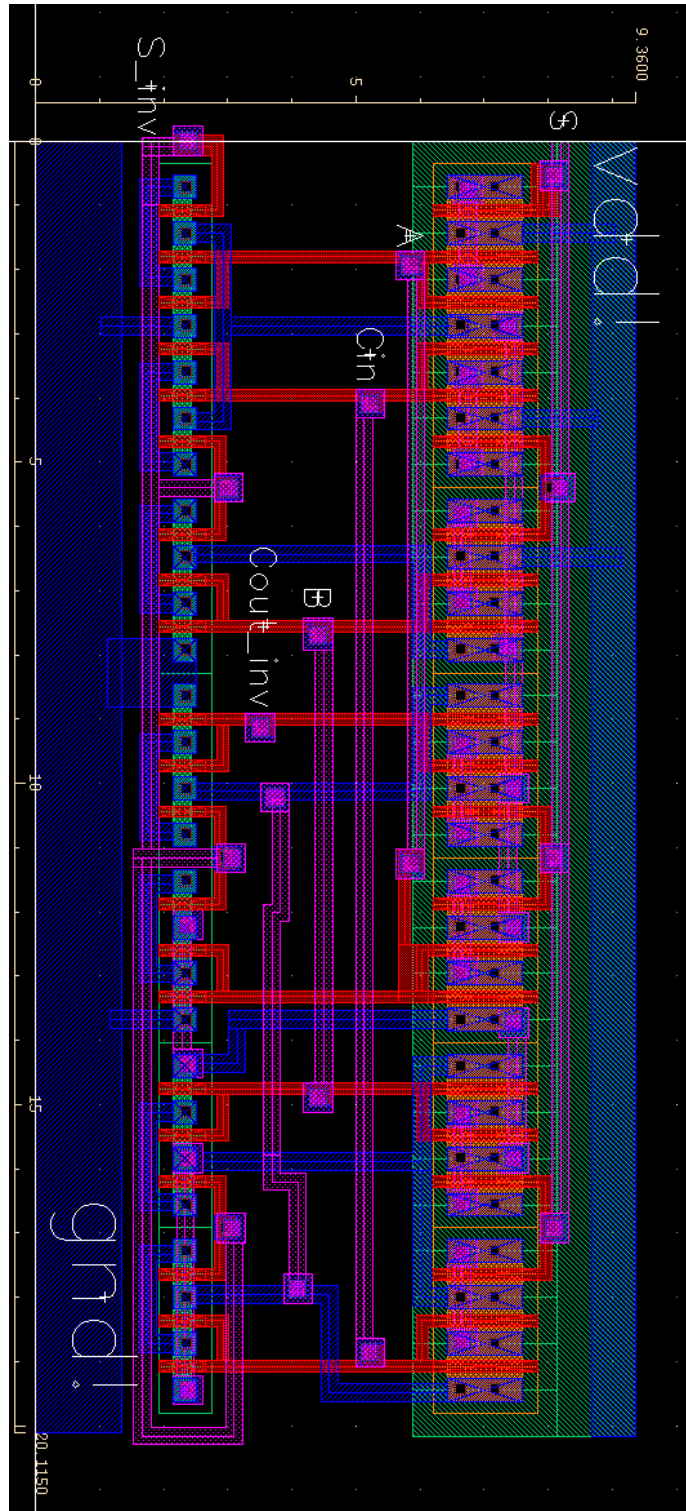
B.5.a.i. Sleepy stack approach Full Adder Cout' Schematic



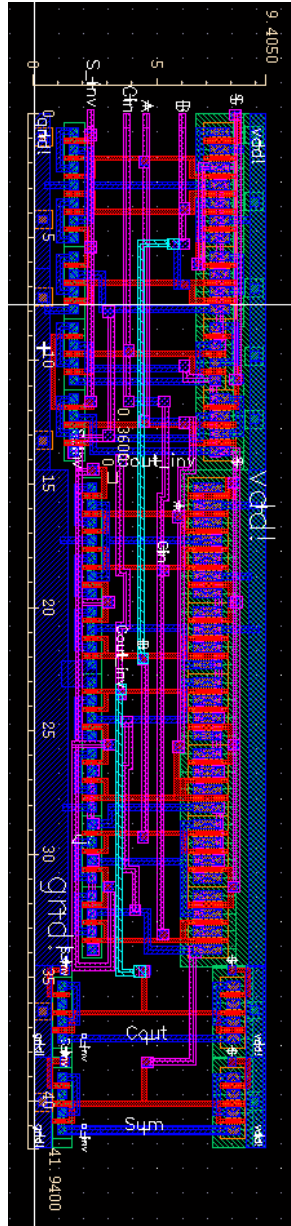
B.5.a.ii. Sleepy stack approach Full Adder Sum' Schematic



B.5.b.i. Sleepy stack approach Full Adder Cout' Layout



B.5.b.ii. Sleepy stack approach Full Adder Sum' Layout



B.5.b.iii. Sleepy stack approach Full Adder Layout

TSMC 0.18μ	Propagation delay (s)	Static Power (W)	Dynamic Power (W)	Area (μ ²)
Base case	6.97E-10	3.87E-10	1.51E-04	138.00
Stack	1.70E-09	2.24E-10	1.30E-04	186.00
Sleep	9.43E-10	1.10E-10	1.55E-04	186.00
ZigZag	9.45E-10	5.49E-11	1.43E-04	166.00
Sleepy Stack	1.36E-09	1.58E-10	1.31E-04	396.00
Sleep (dual Vth)	1.26E-09	1.86E-11	1.59E-04	186.00
ZigZag (dual Vth)	1.26E-09	1.21E-11	1.43E-04	166.00
Sleepy Stack (dual Vth)	1.73E-09	3.83E-11	1.21E-04	396.00

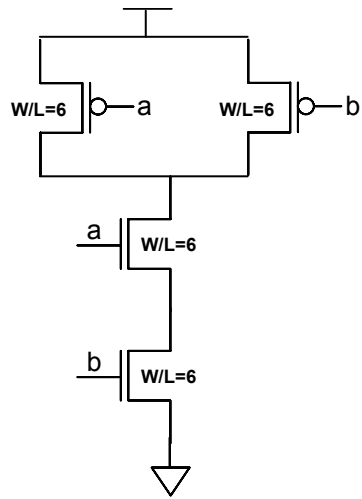
Berkeley 0.18μ	Propagation delay (s)	Static Power (W)	Dynamic Power (W)	Area (μ ²)
Base case	5.07E-10	3.04E-08	1.41E-04	138.00
Stack	1.50E-09	2.96E-09	1.21E-04	186.00
Sleep	6.79E-10	4.51E-09	1.46E-04	186.00
ZigZag	6.83E-10	2.51E-09	1.35E-04	166.00
Sleepy Stack	1.18E-09	4.30E-09	1.27E-04	396.00
Sleep (dual Vth)	9.38E-10	1.33E-11	1.53E-04	186.00
ZigZag (dual Vth)	9.53E-10	8.12E-12	1.37E-04	166.00
Sleepy Stack (dual Vth)	1.63E-09	3.51E-11	1.18E-04	396.00

Berkeley 0.13μ	Propagation delay (s)	Static Power (W)	Dynamic Power (W)	Area (μ ²)
Base case	4.15E-10	2.40E-08	6.10E-05	79.18
Stack	1.21E-09	9.69E-10	5.20E-05	106.72
Sleep	5.46E-10	1.98E-09	6.19E-05	106.72
ZigZag	5.43E-10	1.25E-09	5.83E-05	95.25
Sleepy Stack	9.35E-10	1.63E-09	5.42E-05	227.21
Sleep (dual Vth)	7.53E-10	6.96E-12	6.47E-05	106.72
ZigZag (dual Vth)	7.56E-10	1.66E-12	5.90E-05	95.25
Sleepy Stack (dual Vth)	1.21E-09	2.22E-11	4.94E-05	227.21

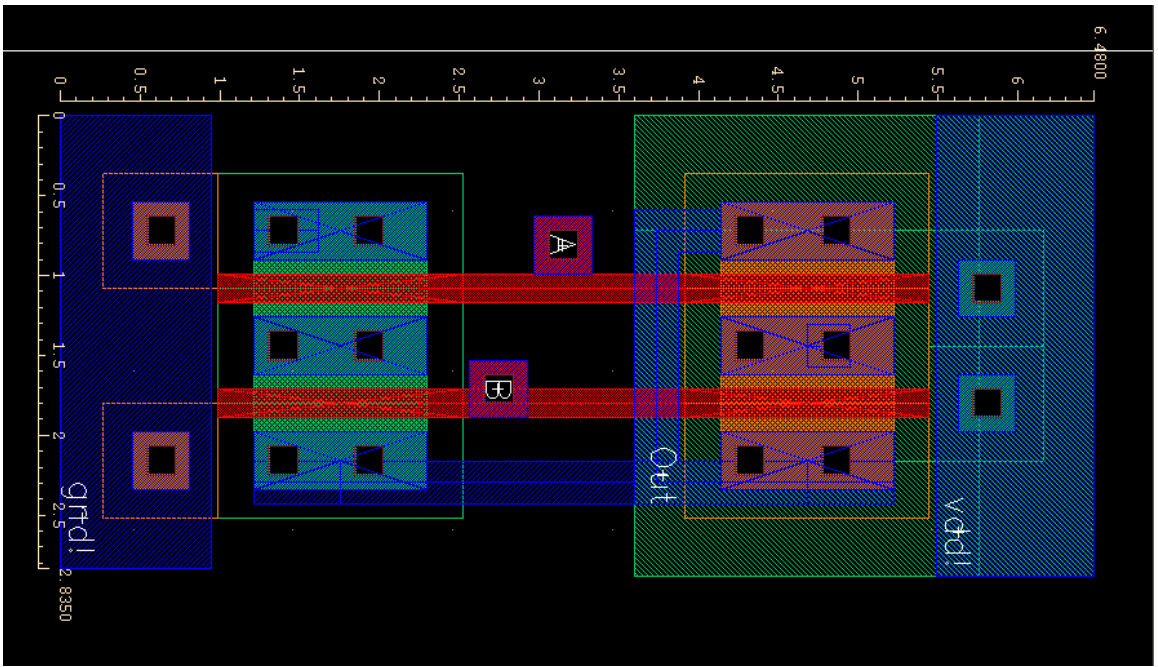
Berkeley 0.10μ	Propagation delay (s)	Static Power (W)	Dynamic Power (W)	Area (μ ²)
Base case	3.08E-10	9.75E-08	3.68E-05	46.85
Stack	8.95E-10	3.20E-09	3.00E-05	63.15
Sleep	4.13E-10	5.26E-09	3.73E-05	63.15
ZigZag	4.17E-10	3.23E-09	3.54E-05	56.36
Sleepy Stack	7.01E-10	5.05E-09	3.19E-05	134.44
Sleep (dual Vth)	5.55E-10	5.72E-12	3.85E-05	63.15
ZigZag (dual Vth)	5.62E-10	4.94E-12	3.55E-05	56.36
Sleepy Stack (dual Vth)	9.14E-10	2.38E-11	2.92E-05	134.44

Berkeley 0.07μ	Propagation delay (s)	Static Power (W)	Dynamic Power (W)	Area (μ ²)
Base case	2.91E-10	1.81E-07	1.52E-05	22.96
Stack	8.89E-10	9.25E-09	1.24E-05	30.94
Sleep	4.11E-10	1.69E-08	1.54E-05	30.94
ZigZag	4.06E-10	1.20E-08	1.47E-05	27.62
Sleepy Stack	6.79E-10	1.50E-08	1.31E-05	65.88
Sleep (dual Vth)	6.20E-10	3.31E-12	1.61E-05	30.94
ZigZag (dual Vth)	6.15E-10	4.92E-12	1.47E-05	27.62
Sleepy Stack (dual Vth)	1.03E-09	1.88E-11	1.22E-05	65.88

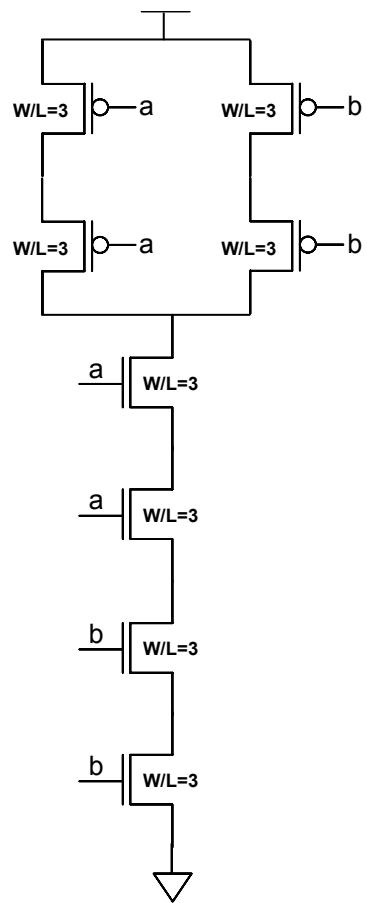
B.6. Adder data



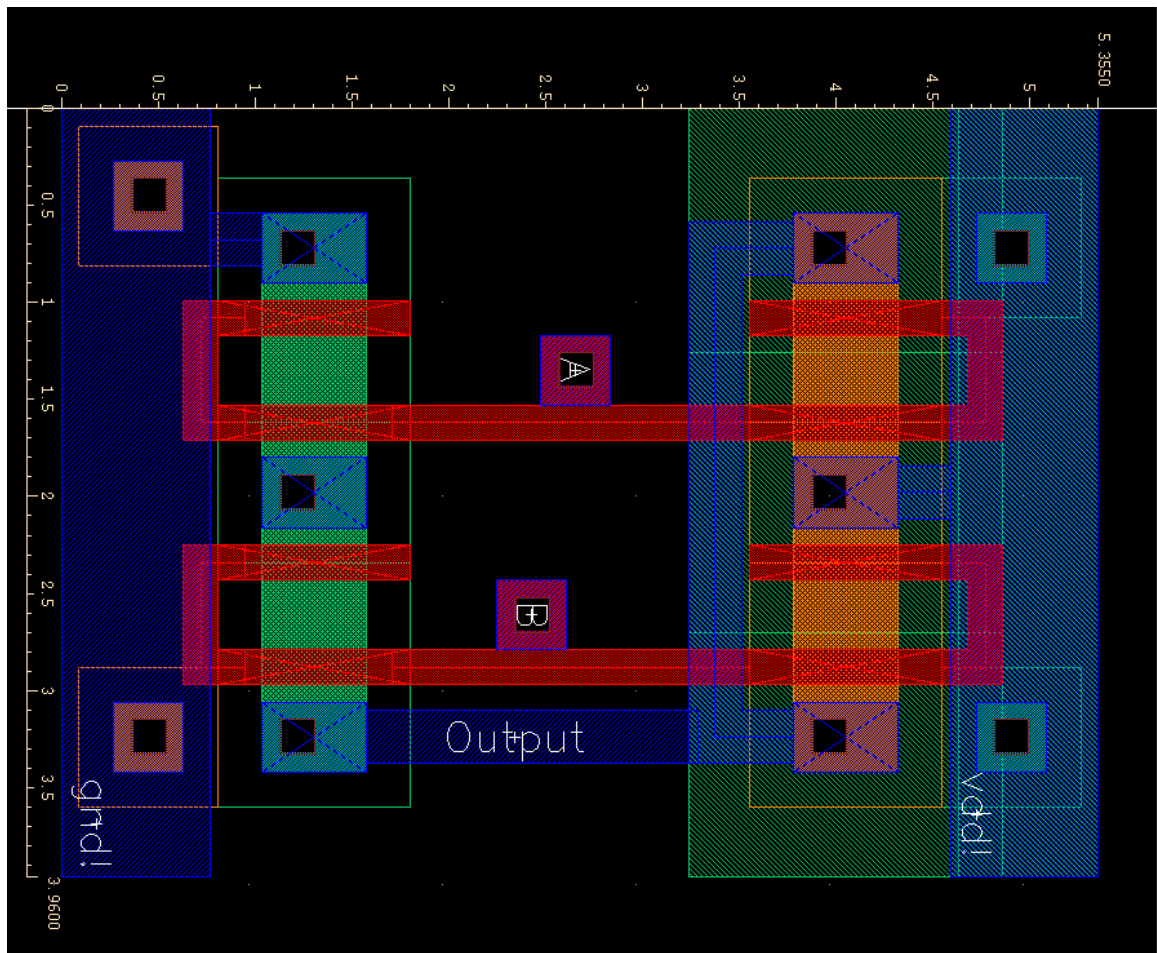
C.1.a. Base case NAND schematic



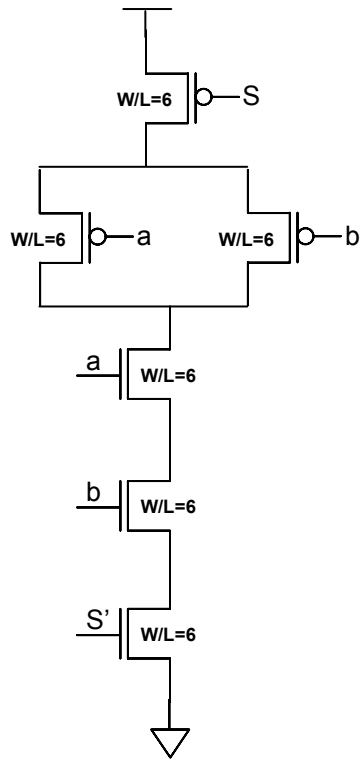
C.1.b. Base case NAND layout



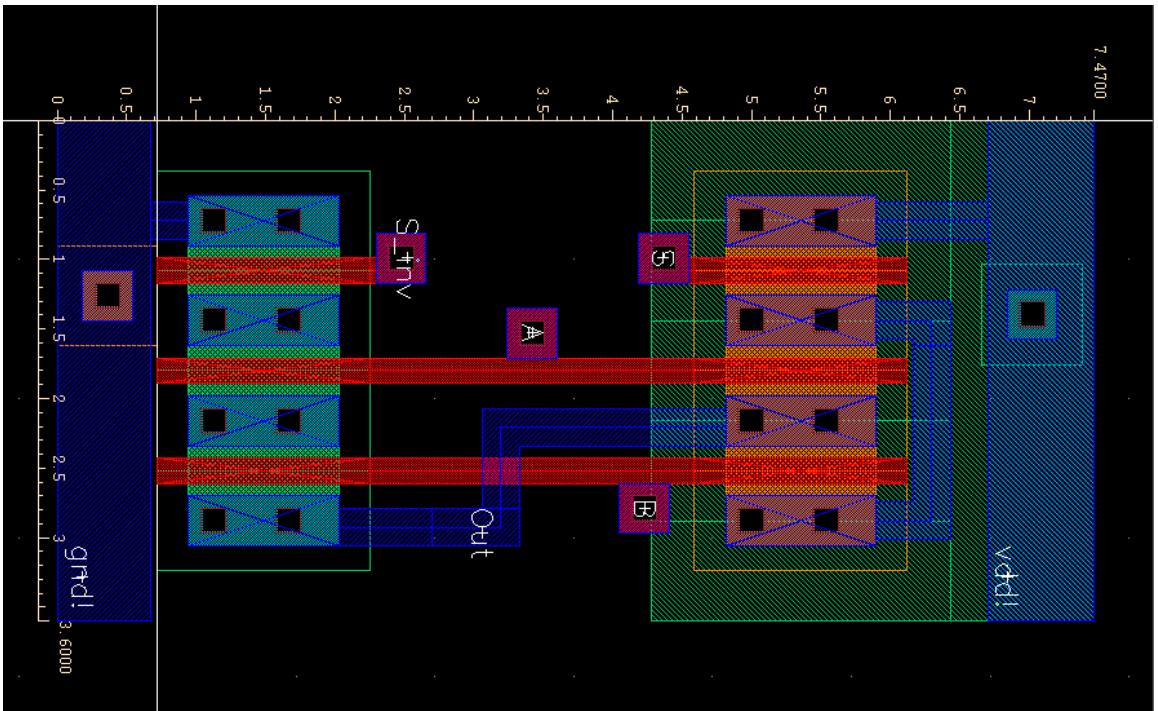
C.2.a. Stack approach NAND Schematic



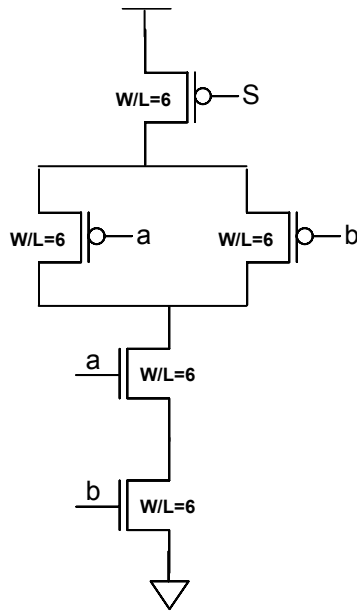
C.2.b. Stack approach NAND Layout



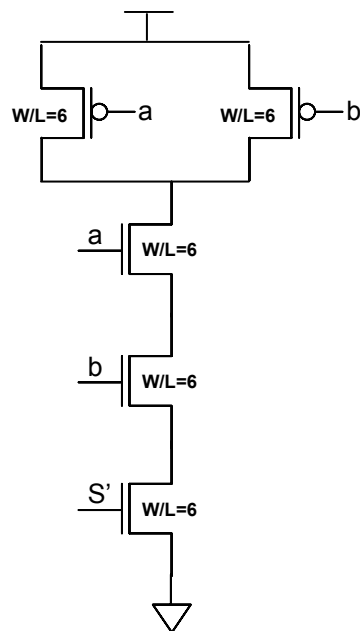
C.3.a. Sleep approach NAND Schematic



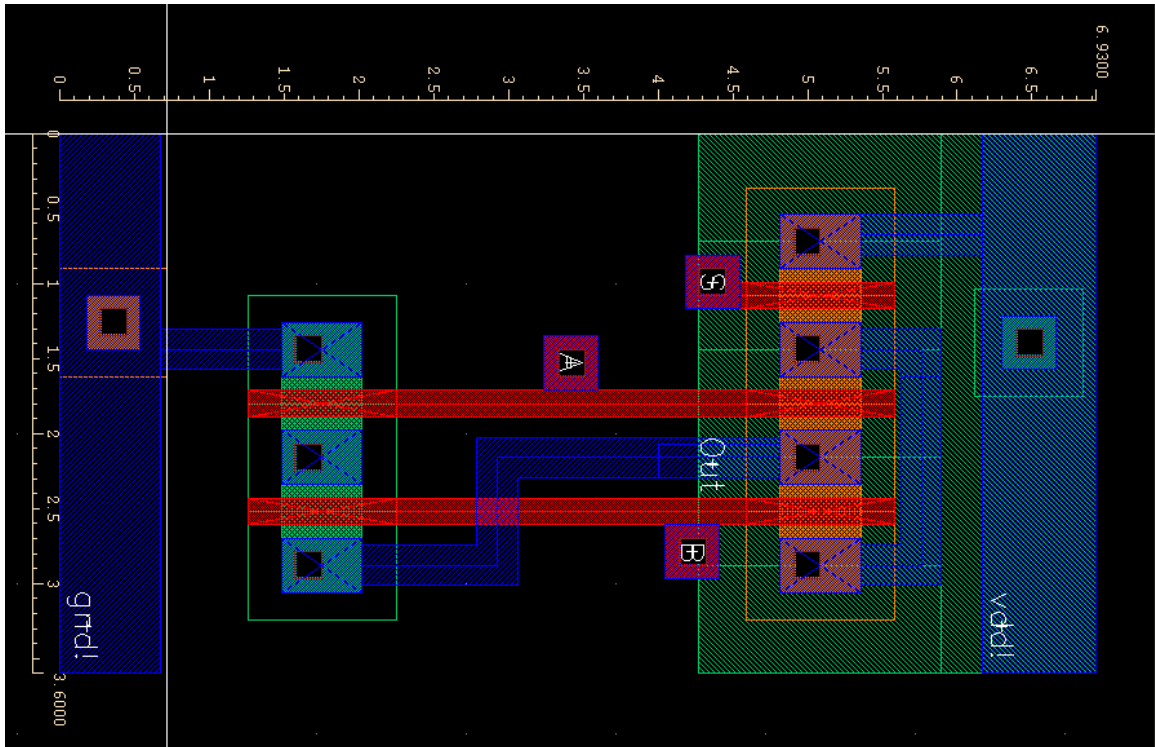
C.3.b. Sleep approach NAND Layout



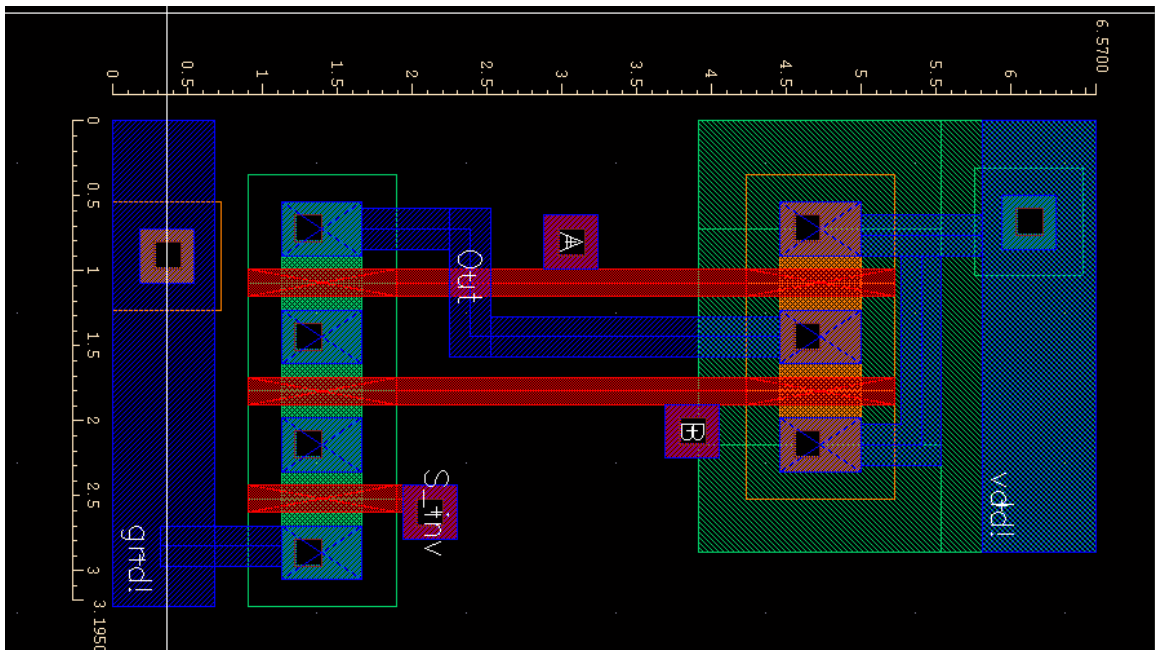
C.4.a.i. Zigzag approach NAND Schematic, PMOS sleep



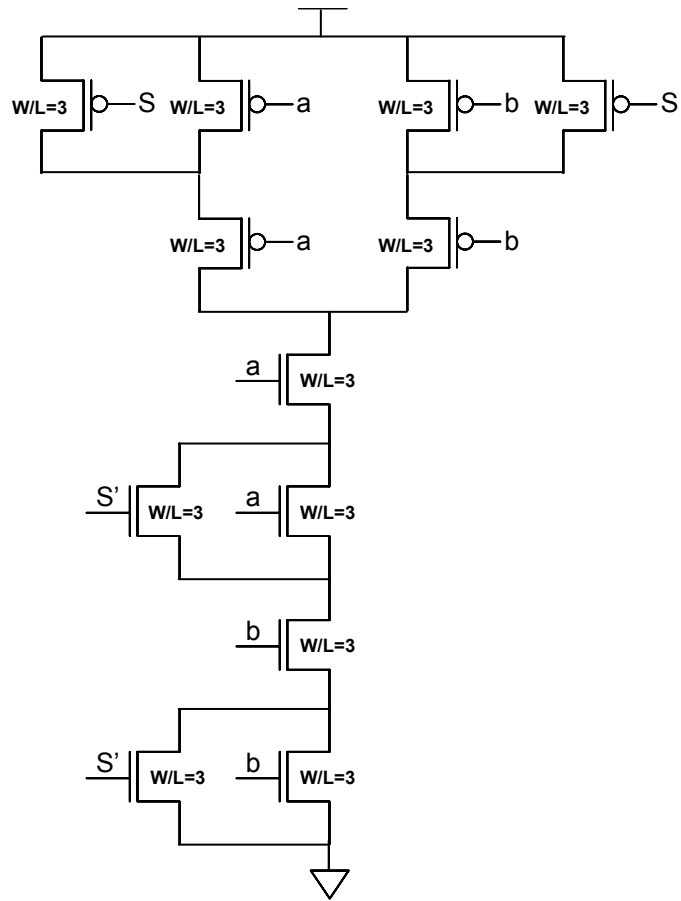
C.4.a.ii. Zigzag approach NAND Schematic, NMOS sleep



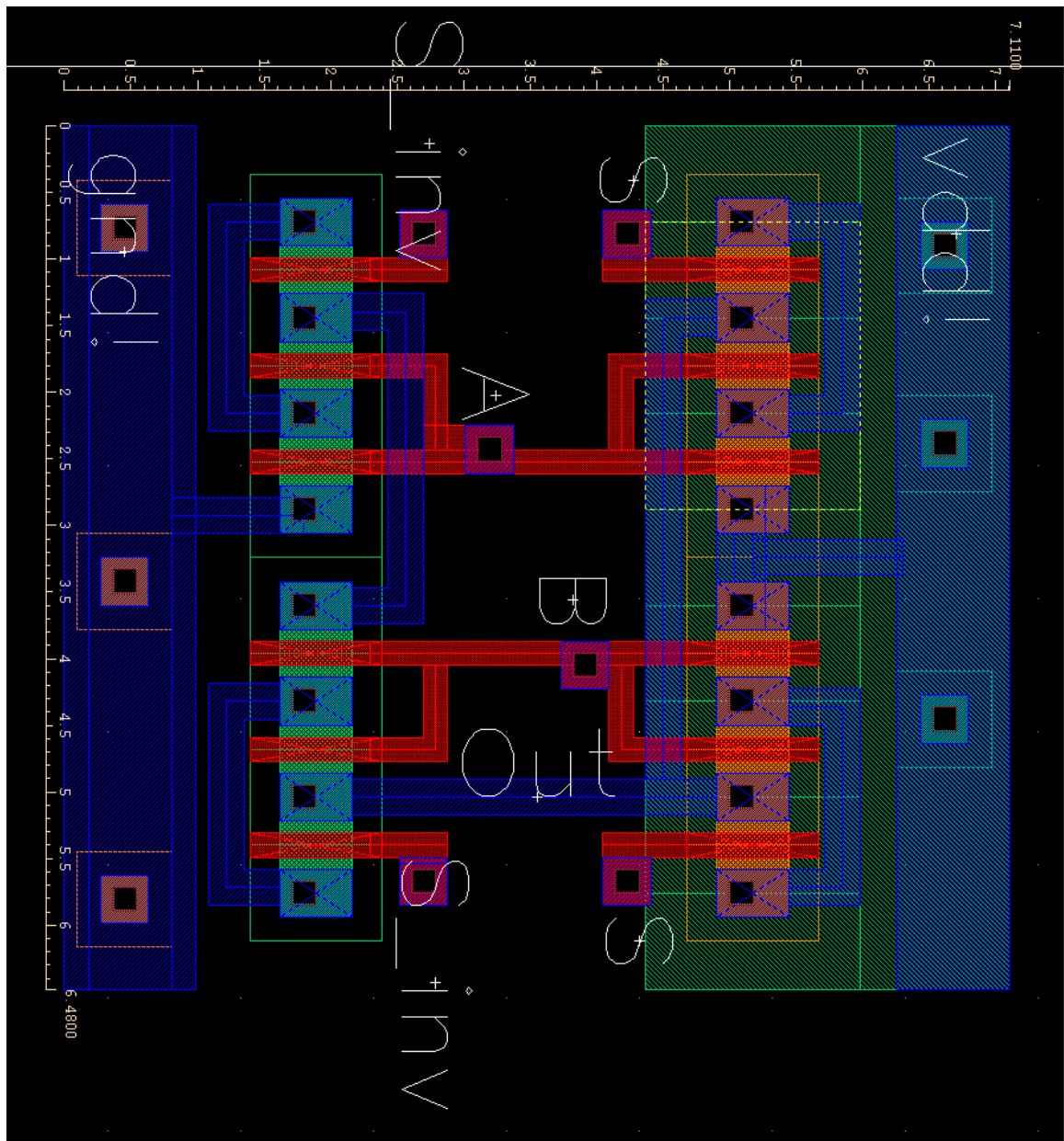
C.4.b.i. Zigzag approach NAND Layout, PMOS sleep



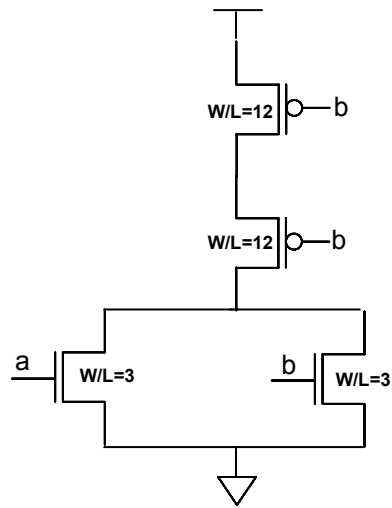
C.4.b.ii. Zigzag approach NAND Layout, NMOS sleep



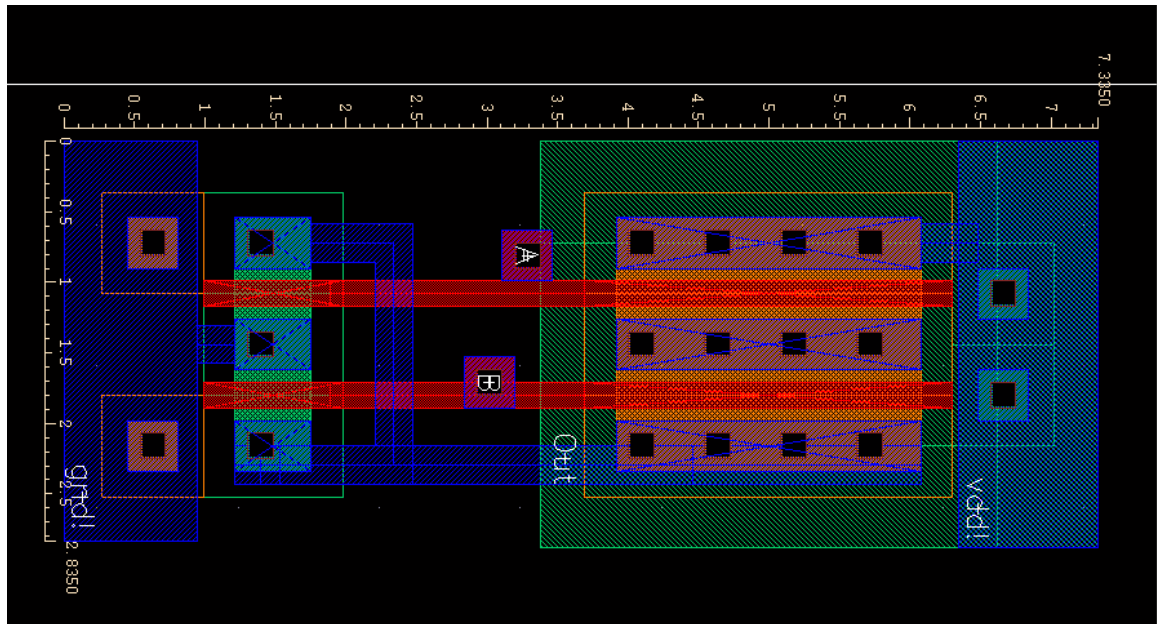
C.5.a. Sleepy stack approach NAND Schematic



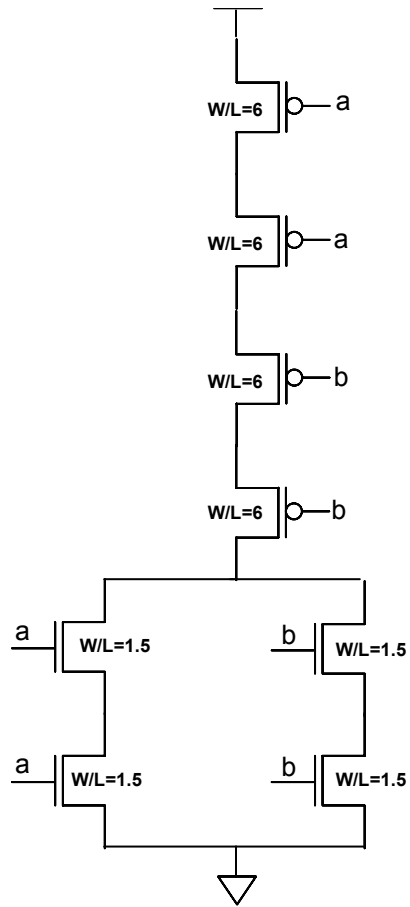
C.5.b. Sleepy stack approach NAND Layout



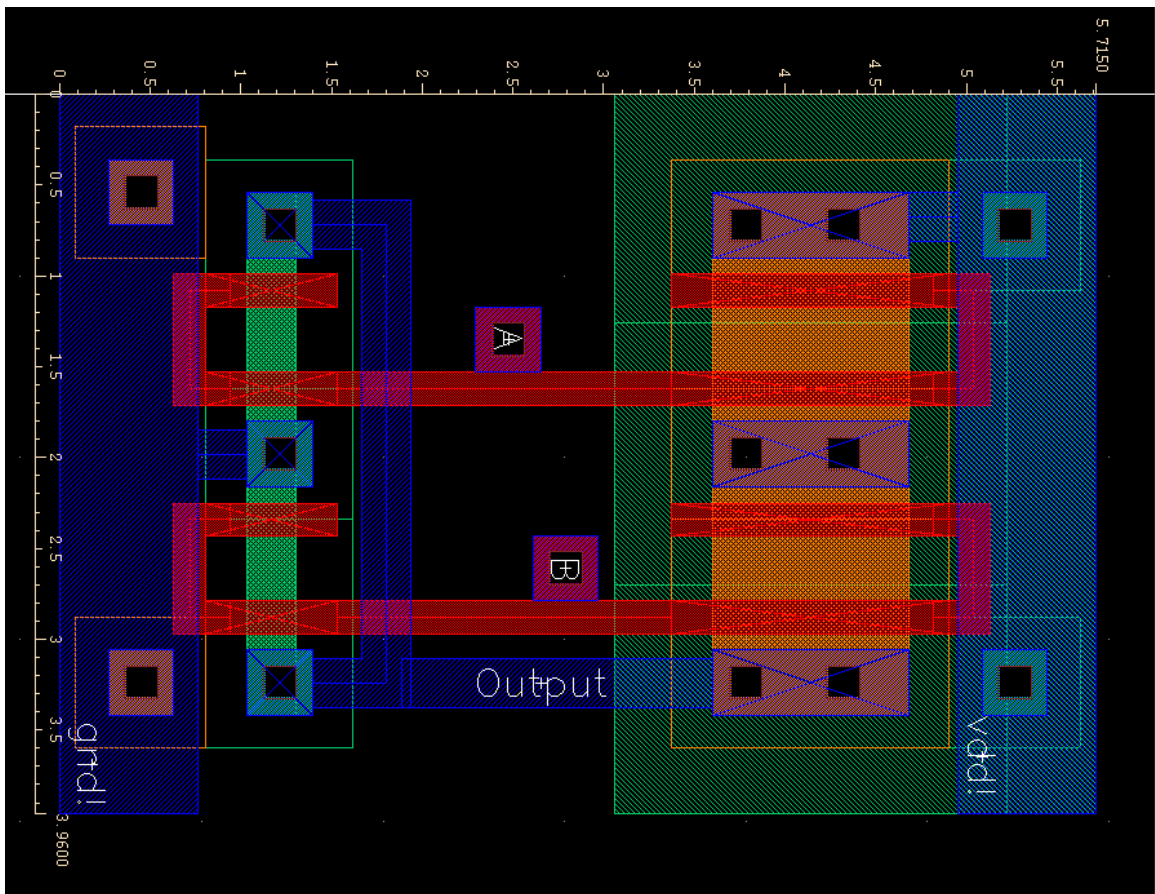
D.1.a. Base case NOR Schematic



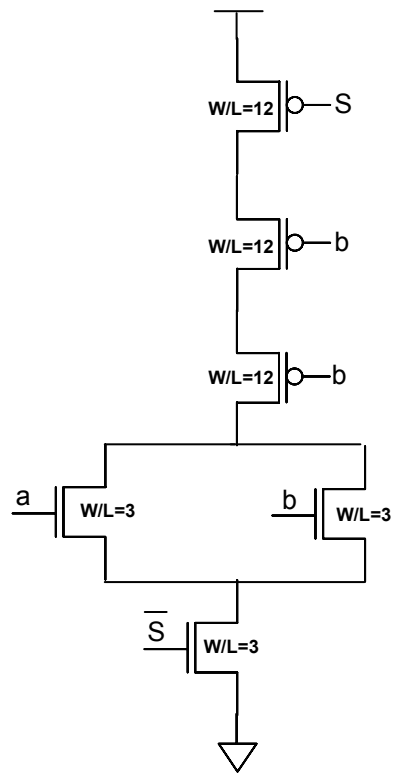
D.1.b. Base case NOR Layout



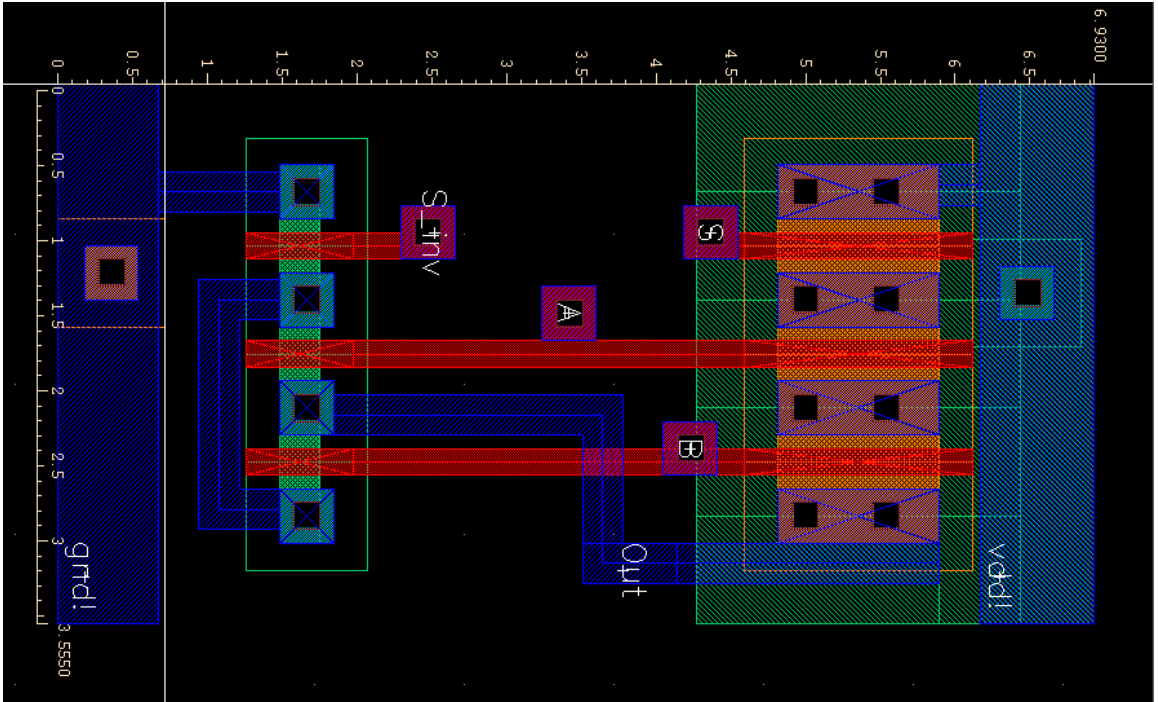
D.2.a. Stack approach NOR Schematic



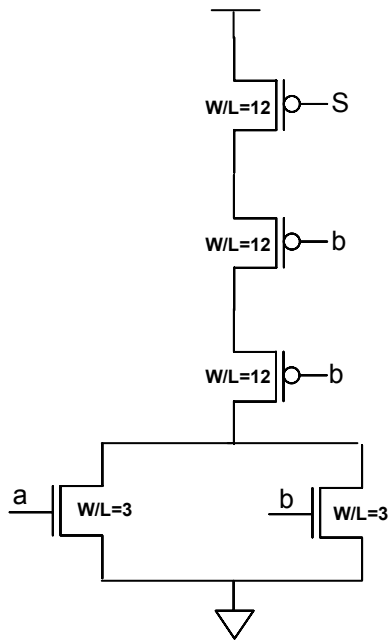
D.2.b. Stack approach NOR Layout



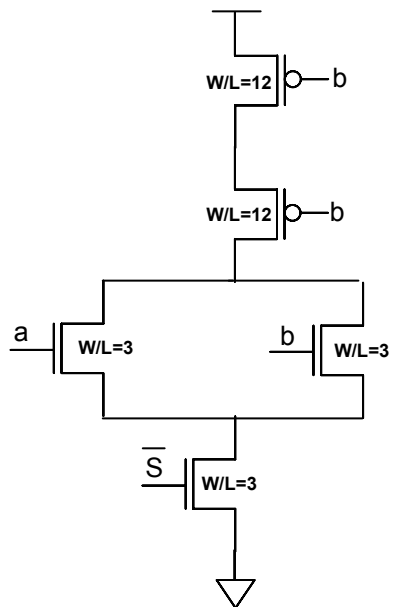
D.3.a. Sleep approach NOR Schematic



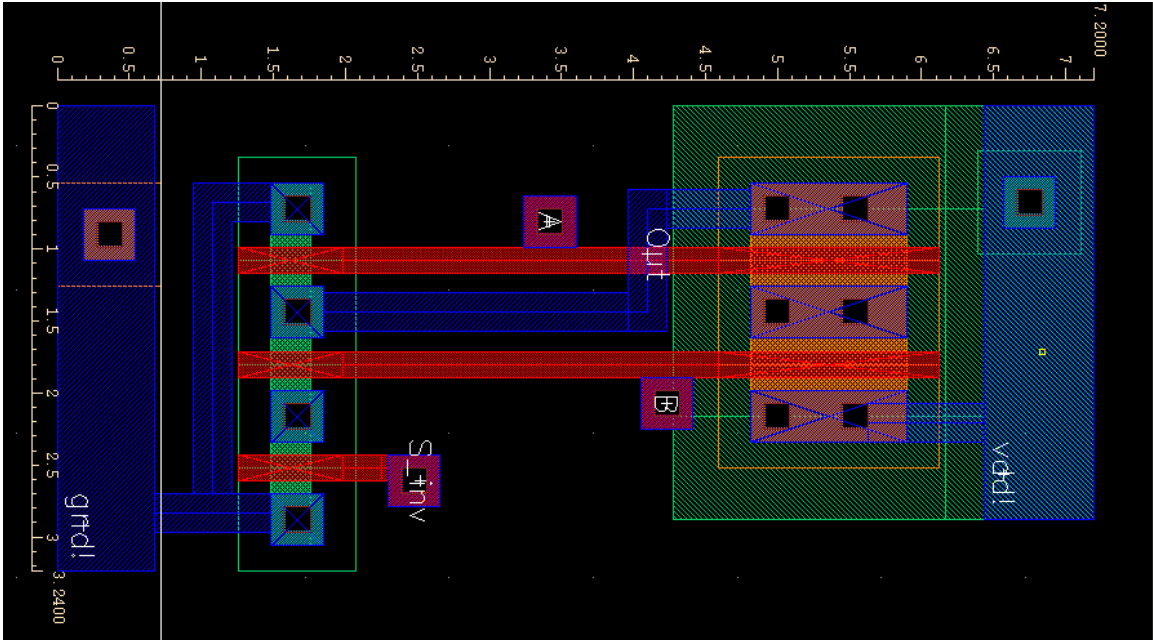
D.3.b. Sleep approach NOR Layout



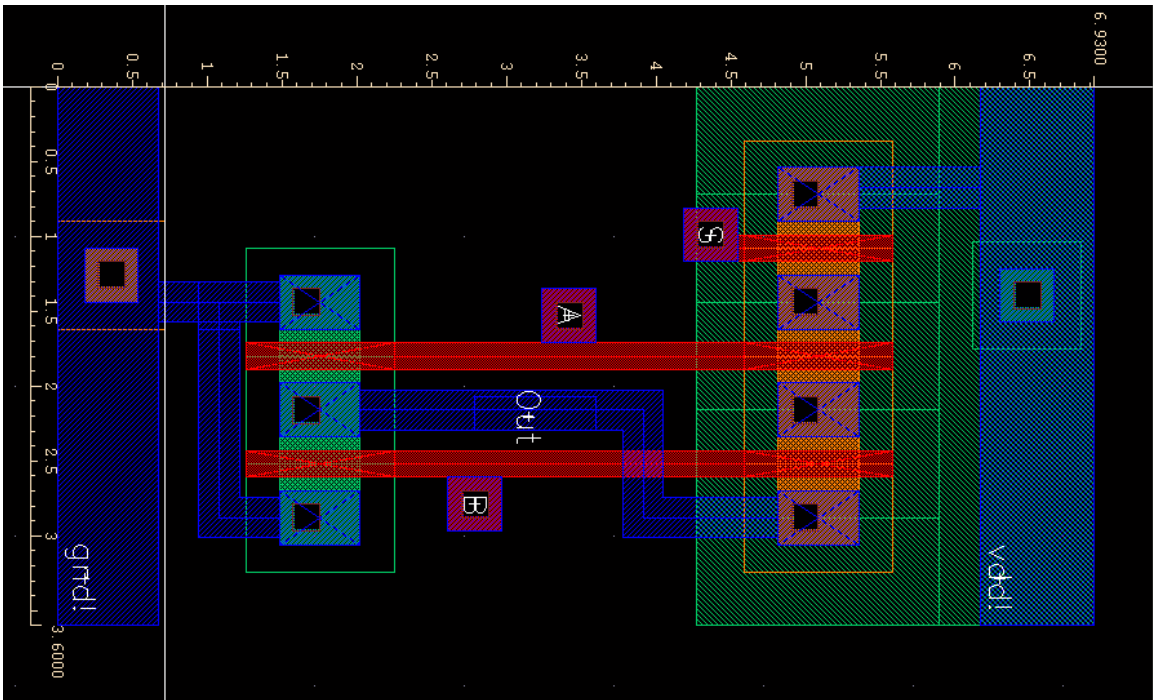
D.4.a.i. Zigzag approach NOR Schematic, PMOS Sleep



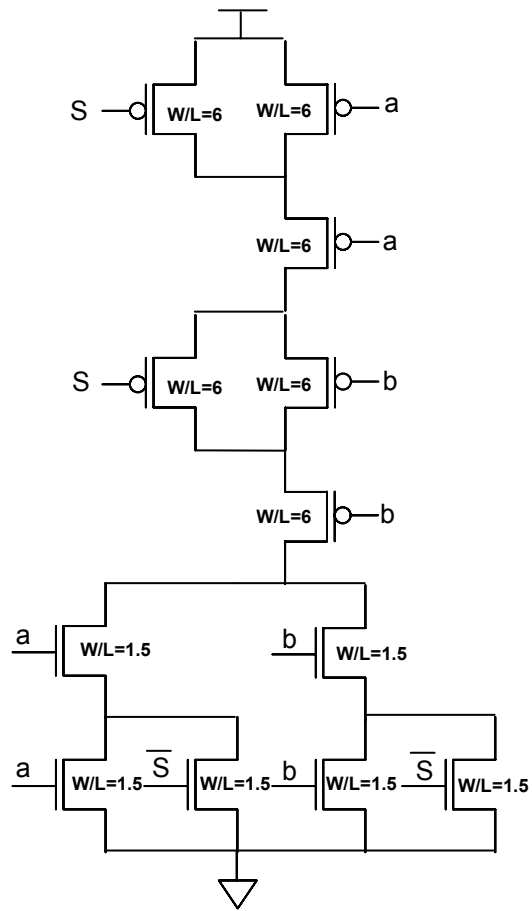
D.4.a.ii. Zigzag approach NOR Schematic, PMOS Sleep



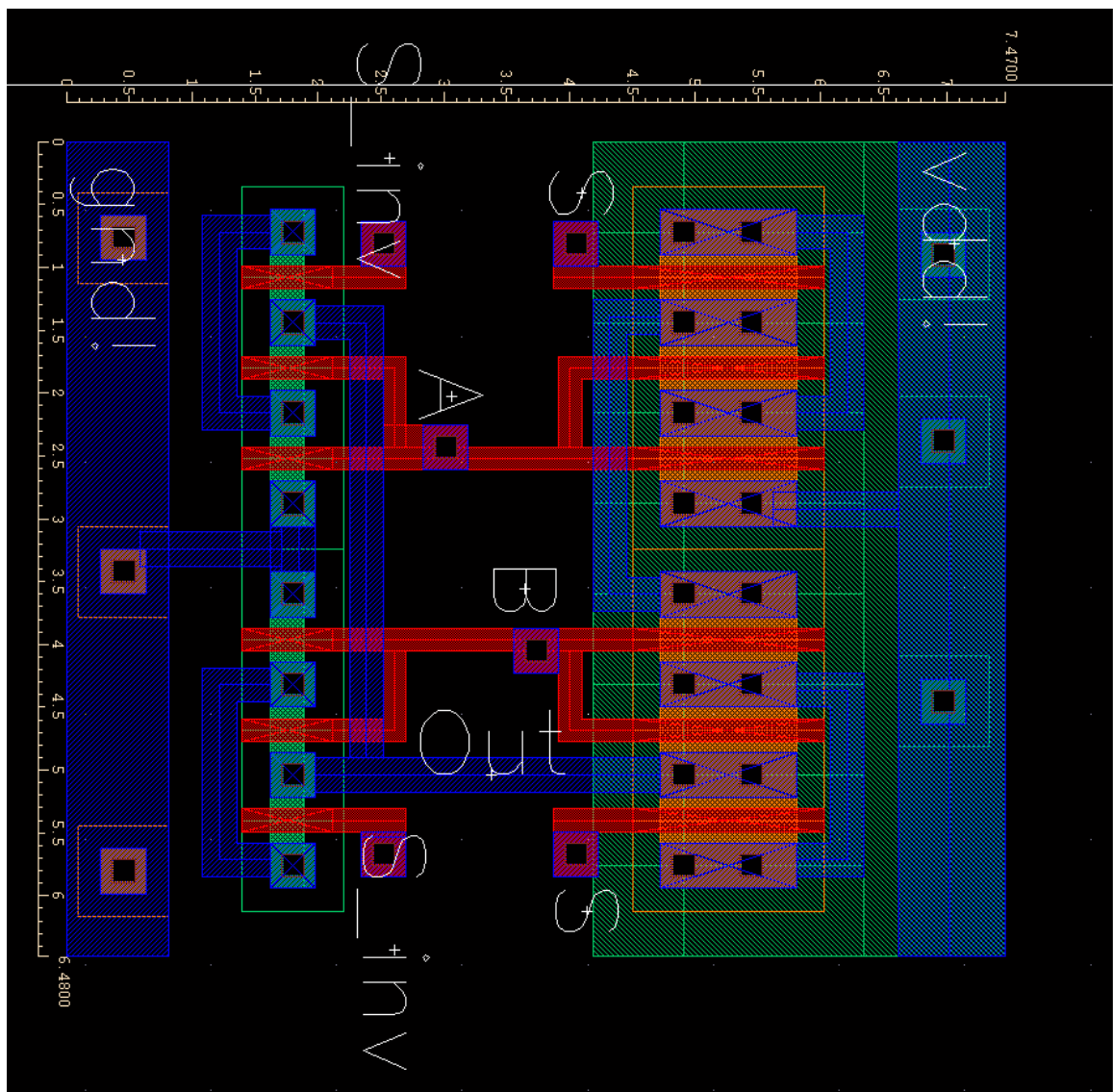
D.4.b.i Zigzag approach NOR Layout, NMOS Sleep



D.4.b.ii. Zigzag approach NOR Layout, PMOS Sleep



D.5.a. Sleepy stack approach NOR Schematic



D.5.b. Sleepy stack approach NOR Layout

TSMC 0.18 μ	Propagation delay (s)	Static Power (W)	Dynamic Power (W)	Area (μ^2)
Base case	2.58E-10	2.89E-10	4.07E-05	301.60
Stack	7.26E-10	4.87E-11	3.45E-05	345.06
Sleep	3.63E-10	7.71E-11	3.40E-05	445.50
ZigZag	5.62E-10	4.75E-11	3.60E-05	447.00
Sleepy Stack	5.62E-10	8.31E-11	3.60E-05	753.40
Sleep (dual Vth)	4.87E-10	6.39E-12	3.47E-05	445.50
ZigZag (dual Vth)	7.41E-10	2.61E-14	3.37E-05	447.00
Sleepy Stack (dual Vth)	7.41E-10	3.67E-12	3.37E-05	753.40

Berkeley 0.18 μ	Propagation delay (s)	Static Power (W)	Dynamic Power (W)	Area (μ^2)
Base case	1.77E-10	2.23E-08	3.69E-05	301.60
Stack	5.50E-10	1.55E-09	3.06E-05	345.06
Sleep	2.39E-10	2.81E-09	3.06E-05	445.50
ZigZag	4.38E-10	1.49E-09	3.27E-05	447.00
Sleepy Stack	4.38E-10	2.63E-09	3.27E-05	753.40
Sleep (dual Vth)	3.36E-10	8.69E-12	3.16E-05	445.50
ZigZag (dual Vth)	5.76E-10	3.98E-13	3.04E-05	447.00
Sleepy Stack (dual Vth)	5.76E-10	3.42E-12	3.04E-05	753.40

Berkeley 0.13 μ	Propagation delay (s)	Static Power (W)	Dynamic Power (W)	Area (μ^2)
Base case	1.48E-10	1.84E-08	1.64E-05	173.05
Stack	4.71E-10	9.02E-10	1.38E-05	197.98
Sleep	2.07E-10	2.59E-09	1.36E-05	255.61
ZigZag	3.59E-10	1.48E-09	1.44E-05	256.47
Sleepy Stack	3.59E-10	1.58E-09	1.44E-05	432.27
Sleep (dual Vth)	2.87E-10	6.60E-12	1.40E-05	255.61
ZigZag (dual Vth)	4.86E-10	1.41E-12	1.37E-05	256.47
Sleepy Stack (dual Vth)	4.86E-10	2.61E-12	1.37E-05	432.27

Berkeley 0.10 μ	Propagation delay (s)	Static Power (W)	Dynamic Power (W)	Area (μ^2)
Base case	1.11E-10	8.62E-08	1.02E-05	102.40
Stack	3.51E-10	2.18E-09	8.03E-06	117.15
Sleep	1.57E-10	5.48E-09	8.39E-06	151.25
ZigZag	2.70E-10	3.16E-09	8.51E-06	151.76
Sleepy Stack	2.70E-10	3.97E-09	8.51E-06	255.78
Sleep (dual Vth)	2.12E-10	5.62E-12	8.50E-06	151.25
ZigZag (dual Vth)	3.59E-10	3.97E-12	7.95E-06	151.76
Sleepy Stack (dual Vth)	3.59E-10	5.46E-12	7.95E-06	255.78

Berkeley 0.07 μ	Propagation delay (s)	Static Power (W)	Dynamic Power (W)	Area (μ^2)
Base case	1.05E-10	1.72E-07	4.35E-06	50.17
Stack	3.39E-10	8.63E-09	3.43E-06	57.40
Sleep	1.56E-10	2.24E-08	3.66E-06	74.11
ZigZag	2.58E-10	1.41E-08	3.64E-06	74.36
Sleepy Stack	2.58E-10	1.51E-08	3.64E-06	125.33
Sleep (dual Vth)	2.35E-10	5.03E-12	3.73E-06	74.11
ZigZag (dual Vth)	3.97E-10	7.54E-12	3.43E-06	74.36
Sleepy Stack (dual Vth)	3.97E-10	8.19E-12	3.43E-06	125.33

E. 4-1 MUX data